



THÈSE DE DOCTORAT

SPÉCIALITÉ : PHYSIQUE

*École Doctorale « Sciences et Technologies de l'Information, des
Télécommunications et des Systèmes »*

Présentée par :

Nicolas COLSON

Sujet :

Mécanismes d'adaptation autonome pour la radio cognitive

Soutenue le 6 mai 2009 devant les membres du jury :

M.	François CHARPILLET	(Rapporteur)
M.	Bruce DENBY	(Rapporteur)
M.	Lionel HUSSON	(Co-directeur de thèse)
M.	Apostolos KOUNTOURIS	(Examineur)
M.	Jacques LEWINER	(Examineur)
M.	Ali MOHAMMAD DJAFARI	(Examineur)
Mme	Armelle WAUTIER	(Directeur de thèse)

Résumé

La radio cognitive apparaît comme une solution naturelle aux problèmes d'échelle et de complexité résultant de la grande popularité des communications sans fil et de l'évolution des technologies radio. Une radio cognitive est un agent intelligent capable de s'adapter à son contexte opérationnel pour (1) respecter le cadre de régulation contrôlant l'accès au spectre, (2) satisfaire les besoins de l'utilisateur en termes de qualité de service, et (3) assurer une gestion optimisée des ressources disponibles (radios, réseaux et matérielles). Ce nouveau paradigme est directement lié au développement de l'intelligence embarquée, objet de ce travail de thèse.

Dans ce mémoire de thèse, nous détaillons la conception d'un moteur cognitif structurant les opérations de raisonnement et d'apprentissage nécessaires à la supervision du processus de reconfiguration dynamique. La solution proposée suit une approche originale basée sur une modélisation qualitative du problème de conception à la volée.

Le moteur cognitif raisonne de manière autonome en s'appuyant sur les relations d'ordre définies par deux échelles déduites dynamiquement en fonction du contexte. Il navigue le long de ces deux échelles pour rechercher une configuration adaptée avec un souci d'efficacité et d'optimalité. Il exploite ses capacités de prédiction afin d'estimer l'impact de l'environnement radio sur les performances des configurations disponibles (nombre d'erreurs de transmission tolérables). Il évalue les configurations compatibles avec le service à l'aide d'un système de notation calculant la satisfaction des alternatives vis-à-vis des objectifs d'optimisation (réduire la consommation d'énergie, maximiser le débit de transmission). Si nécessaire, une expérience de conception est déclenchée afin d'explorer l'espace de conception en ligne. Les connaissances mémorisées sont alors mises à l'épreuve dans le but de se rapprocher du comportement optimal. Le plan d'expérience est construit dynamiquement en fonction des retours de l'environnement. Le moteur cognitif profite également de connaissances expertes embarquées pour limiter les risques d'expérimentation. Il accumule de l'expérience progressivement et il apprend à recourir de moins en moins souvent à l'expérimentation pour exploiter ses connaissances devenues fiables.

Nous avons testé notre approche sur deux études de cas de conception cognitive d'un lien radio. Les résultats obtenus confirment la pertinence des mécanismes cognitifs proposés. Le moteur cognitif parvient en effet à trouver la solution optimale pour la grande majorité des problèmes traités (95% en moyenne) et il devient de plus en plus efficace dans sa recherche d'une configuration adaptée.

Notre méthode combine efficacement la puissance de systèmes d'apprentissage avec des connaissances issues de l'expertise en ingénierie des télécommunications. Le moteur cognitif est conçu avec une grande autonomie décisionnelle grâce à des capacités de raisonnement, d'exploration en ligne et d'apprentissage incrémental. De plus, nous avons proposé des mécanismes avancés pour améliorer son comportement afin d'en faire une solution cognitive effective pour de nombreux problèmes de conception dynamique.

Mots-clés: Radio cognitive, intelligence artificielle, prise de décision autonome, conception du lien radio, systèmes connexionnistes évolutifs, automates d'apprentissage.

Abstract

Cognitive radio appears as a natural solution to the complexity and scaling issues stemming from the increasing popularity of wireless communications and the radio technologies evolution. A cognitive radio is a smart agent adapting its operation to its context in order to (1) respect the regulation constraints on spectrum access, (2) satisfy the user needs in terms of Quality of Service, and (3) ensure an optimized management of the available resources (e.g. network, radio and device hardware resources). This new paradigm is directly linked to the development of embedded intelligence, subject of this PhD thesis.

This dissertation details the design of a cognitive engine structuring the reasoning and learning operations required for the supervision of the dynamic reconfiguration process. The proposed solution follows an original approach based on a qualitative modeling of the cognitive design problem.

The cognitive engine reasons autonomously by relying on the order relationships defined by two scales dynamically set according to the context. It navigates along these two scales to search for an adapted configuration with a concern on efficiency and optimality. It exploits its predictive capabilities to estimate the impact of the radio environment on the available configurations performance (number of transmission errors tolerable). It evaluates the configurations compatible with the service thanks to a grading system assessing the alternatives satisfaction with regard to the optimization objectives (reduce the energy consumption, maximize the data rate). If necessary, a design experience is triggered for exploring the design space online. The memorized knowledge is then put to test in order to come closer to the optimal behavior. The experiment is designed dynamically according to the environment feedbacks. The cognitive engine takes also advantage of embedded expert knowledge to limit the experimental risks. It accumulates experience progressively and it learns to appeal less often to experimentation in order to exploit its knowledge that became reliable.

We have tested our approach on two case studies of cognitive waveform design. The results obtained confirm the pertinence of the proposed cognitive mechanisms. The cognitive engine manages to find the optimal solution for most of the processed problems (95% in average) and it keeps increasing its efficiency in the search of an appropriate configuration.

Our modeling method combines efficiently the power of learning systems with knowledge stemming from telecommunications expertise. The cognitive engine is designed with a great decision autonomy through its ability to reason, explore online and learn incrementally. Moreover, we have proposed advanced mechanisms to optimize its behavior in order to make it an effective cognitive solution for many design problems.

Keywords: Cognitive Radio, Artificial Intelligence, Autonomous Decisions Making, Waveform Design, Evolving Connectionist Systems, Learning Automata.

Remerciements

Une période s'achève et il est temps de remercier toutes les personnes qui m'ont accompagnées pendant ces trois années particulièrement formatrices tant sur le plan humain que professionnel.

Je tiens tout d'abord à remercier Apostolos Kountouris d'avoir cru en moi et de m'avoir proposé un stage puis cette thèse. Le pari n'était pourtant pas gagné au début de cette aventure, au regard de mes connaissances limitées en télécommunications et en intelligence artificielle. Il a su voir ma formation pluridisciplinaire comme une force plutôt qu'une faiblesse et je lui en suis reconnaissant.

Je remercie mes directeurs de thèse, Armelle Wautier et Lionel Husson, pour leur expertise qu'ils ont partagée au travers de nos réunions ainsi que pour leur relecture attentive du manuscrit. La présentation du document a grandement bénéficié de leurs commentaires.

J'adresse tous mes remerciements à l'ensemble des membres du jury pour l'intérêt qu'ils ont porté à mes travaux. A M. François Charpillet, directeur de recherche à l'INRIA, et M. Bruce Denby, professeur à l'UPMC, pour m'avoir fait l'honneur de rédiger un rapport sur ma thèse. A M. Jacques Lewiner, professeur à l'ESPCI, et M. Ali Mohammad-Djafari, directeur de recherche au CNRS, pour avoir accepté d'évaluer mes travaux en tant qu'examineurs.

Je remercie tous mes collègues d'Orange Labs qui ont contribué à la bonne ambiance de travail tout en se montrant disponibles pour des questions d'ordre scientifique ou administratif. Je pense notamment à mes confrères doctorants, à Mischa, Fred, Benoit et Jean Marc.

Je remercie plus particulièrement Jean Da Rolt Joaquim pour son investissement personnel dans le stage que je lui ai proposé. Je n'ai malheureusement pas eu le temps d'intégrer mes travaux sur la plateforme cognitive que nous avons développée. Mais je garde un excellent souvenir de notre collaboration en raison de son sérieux et de sa disponibilité. Nos échanges fructueux n'ont fait que confirmer mon goût pour le travail en équipe.

Je pense bien évidemment à toute ma famille et tous mes amis sans qui je n'aurais pas tenu lors des moments difficiles. Je les remercie sincèrement pour leur patience face à mes états d'âme, pour leur attitude toujours positive et pour tous les moments de joie partagés, que ce soit dans la vie de tous les jours, en vacances ou en soirées. Mention spéciale pour mon frère Arnaud, mes cousins (Tere, Teva et Terii), les amis rencontrés à Grenoble (Chawki, Tarik, Cédric, Flo et la team du GVUC, Denis et Raïssa) et mes amis parisiens (Ben, François, tout le D29, Dorian, Walter, Abou et Joseph). Vous avez su me donner le sourire très régulièrement et je vous en remercie.

A very special thought goes to Yon for her constant support and care. You helped me overcome my doubts and stress towards the completion of this PhD. I'm not sure if I would have made it without you! I'd also like to address a big thank to your family for the excellent working conditions they provided me with while I was writing my PhD in Tokyo.

Enfin, je remercie mes parents pour la qualité de leur éducation, pour leur confiance et leur soutien sans faille. Je considère qu'ils ont et qu'ils auront toujours une part importante dans mes réussites professionnelles et personnelles.

Table des matières

Table des figures	xi
-------------------	----

Liste des tableaux	xv
--------------------	----

Chapitre 1

Introduction

1.1	Contexte	1
1.1.1	Conception statistique d'un système radio	1
1.1.2	Vers une adaptation dynamique autonome	4
1.1.3	Concept et champs d'application de la radio cognitive	6
1.2	Objectifs de la thèse	7
1.3	Organisation du document	7

Chapitre 2

Intelligence artificielle appliquée aux télécommunications

2.1	Introduction d'un cycle cognitif dans une radio flexible	10
2.2	Conception d'un système de communication cognitif	11
2.2.1	Contexte opérationnel d'une radio	11
2.2.2	Organisation interne d'une radio cognitive	13
2.2.3	Axes de recherche	13
2.3	Conception d'un moteur cognitif	16
2.3.1	Détermination du problème de conception	17
2.3.2	Résolution du problème de conception	18
2.3.3	Synoptique d'un moteur cognitif	19
2.4	Approches cognitives existantes	19
2.4.1	Approche experte	20
2.4.2	Approche exploratoire	22
2.4.3	Approche prédictive	24
2.5	Approche adoptée dans nos travaux	26

Chapitre 3

Démarche de conception du moteur cognitif

3.1	Scénario considéré pour l'évaluation du moteur cognitif	34
3.2	Modélisation du problème de prise de décision autonome	37
3.2.1	Hierarchisation des objectifs et étapes de résolution du problème	37
3.2.2	Introduction de relations d'ordre entre les alternatives	39
3.2.3	Formalisation de la prise de décision comme un problème de classification	41
3.2.4	Avantages du modèle proposé	46
3.3	Codification de l'expertise	48
3.4	Cadre d'apprentissage	52
3.4.1	Discussion sur les différentes formes d'apprentissage	53
3.4.2	Buts de l'apprentissage	55
3.5	Techniques d'intelligence artificielle choisies	56
3.5.1	Pour la prédiction des performances	57
3.5.2	Pour la gestion du dilemme exploration/exploitation	59
3.5.3	Pour la sélection de la configuration la plus adaptée	62

Chapitre 4

Mécanismes cognitifs pour la prise de décision autonome dans un espace de conception totalement ordonné

4.1	Organisation structurelle du moteur cognitif	66
4.2	Description des mécanismes cognitifs	68
4.2.1	Processus d'analyse prédictive	68
4.2.2	Processus de décision	71
4.2.3	Processus d'expérimentation	73
4.2.4	Processus d'apprentissage	73
4.2.5	Algorithme de supervision	76
4.3	Evaluation du moteur cognitif	77
4.3.1	Présentation des indicateurs de performance	77
4.3.2	Présentation des radios de référence	81
4.3.3	Résultats	82

Chapitre 5

Mécanismes cognitifs pour la prise de décision autonome dans un espace de conception partiellement ordonné

5.1	Organisation structurelle du moteur cognitif généralisé	98
5.2	Extension des mécanismes cognitifs	99

5.2.1	Processus d'analyse prédictive	99
5.2.2	Processus d'expérimentation	101
5.2.3	Processus d'apprentissage	108
5.2.4	Algorithme de supervision	111
5.3	Evaluation du moteur cognitif	113
5.3.1	Présentation d'indicateurs de performance supplémentaires	113
5.3.2	Résultats	115

Chapitre 6

Moteur cognitif complet fondé sur les concepts proposés

6.1	Améliorations du moteur à l'aide de mécanismes cognitifs avancés	138
6.1.1	Améliorations par des techniques existantes	138
6.1.2	Mécanismes proposés pour des améliorations supplémentaires	141
6.2	Gestion d'un environnement dynamique	142
6.2.1	Organisation structurelle du moteur cognitif	142
6.2.2	Organisation fonctionnelle du moteur cognitif	143
6.3	Gestion d'un espace de conception à complexité accrue	145
6.3.1	Prise en compte de plusieurs contraintes de performance simultanées . . .	145
6.3.2	Augmentation de la flexibilité des équipements	146
6.4	Application des concepts à l'accès opportuniste au spectre	150

Chapitre 7

Conclusion et perspectives

7.1	Bilan des travaux réalisés	153
7.2	Contributions	156

Annexe A

Modélisation analytique du système de communication reconfigurable

Annexe B

Compléments d'information sur les mécanismes cognitifs

B.1	Processus d'expérimentation généralisé	161
B.2	Résultats supplémentaires	162

Glossaire	165
------------------	------------

Références bibliographiques	167
------------------------------------	------------

Table des figures

1.1	Processus de conception d'un système radio	2
2.1	Cycle cognitif générique	10
2.2	Architecture haut-niveau d'une radio cognitive (extrait de [25])	12
2.3	Synoptique d'un moteur cognitif	19
2.4	Cycle cognitif pour l'approche experte	21
2.5	Cycle cognitif pour l'approche exploratoire	24
2.6	Cycle cognitif pour l'approche prédictive	26
2.7	Cycle cognitif adopté	27
2.8	Partitionnement de l'espace de conception pour différentes qualités du modèle prédictif (a, b, c, d)	29
2.9	Partitionnement de l'espace de conception sans (a) ou avec (b) opération de réduction préalable liée au cadre de régulation et aux contraintes matérielles des objets communicants	31
2.10	Partitionnement de l'espace de conception pour un exemple de contexte favorable (a) ou défavorable (b)	31
2.11	Partitionnement de l'espace de conception pour un moteur cognitif ayant accumulé plus (a) ou moins (b) d'expérience vis-à-vis du problème traité	31
3.1	Scénario de communication considéré pour tester le moteur cognitif	35
3.2	Étapes de la résolution du problème de conception	38
3.3	Interprétation du classement des configurations selon l'échelle de performance	41
3.4	Exemple d'un espace de conception totalement ordonné	43
3.5	Exemple d'un espace de conception partiellement ordonné	43
3.6	Scénarios de prise de décision pour différentes politiques d'optimisation avec ou sans erreurs dans l'échelle de performance	45
3.7	Interprétation du HCR par rapport au nombre de configurations compatibles avec le service	47
3.8	Graphe de robustesse associé au cas de conception totalement ordonné représenté à la figure 3.4	50
3.9	Graphe de robustesse associé au cas de conception partiellement ordonné représenté à la figure 3.5	50
3.10	Graphe de robustesse associé à un espace de conception impliquant toutes les configurations considérées dans le scénario de test	51
3.11	Identification de la radio cognitive comme un problème d'apprentissage par renforcement	54

3.12	Quantification linguistique associée à la variable SNR pour deux ensembles flous (SNR faible, SNR élevé) sans (a) et avec (b) prise en charge des incertitudes . . .	59
3.13	Structure connexionniste du modèle ECF (Evolving Classification Function) [131]	60
4.1	Organisation structurelle du moteur cognitif	67
4.2	Déroulement du processus d'expérimentation pour un espace de conception totalement ordonné	74
4.3	Algorithme de supervision pour un espace de conception totalement ordonné . . .	77
4.4	Scénario de communication adopté pour l'étude de cas considérée	78
4.5	Pourcentage d'expériences de conception n'impliquant aucune mauvaise décision lors de la phase de recherche d'une configuration adaptée (CG_2)	84
4.6	Pourcentage d'expériences de conception se terminant par une décision optimale (CG_3)	85
4.7	Distance moyennée au comportement optimal en termes de rangs dans l'échelle d'optimalité (CG_4)	85
4.8	Nombre moyenné de décisions prises par expérience de conception (CG_5)	87
4.9	Nombre moyenné de mauvaises décisions prises par expérience de conception (CG_6)	87
4.10	Analyse de la réponse du moteur cognitif aux problèmes de conception posés (CG_{7-11})	89
4.11	Taux d'exploration moyenné sur l'ensemble des automates d'apprentissage ($PSSE_1$)	90
4.12	Fiabilité des prédictions du module CAP (FMP_2) comparée à la qualité des décisions prises par le moteur cognitif (FMP_1)	90
4.13	Evolution de la carte d'adéquation avec l'expérience du moteur cognitif (FMP_3)	92
4.14	Evolution du nombre de règles à l'intérieur de la base de connaissances (CMP_1) .	93
5.1	Organisation structurelle du moteur cognitif généralisé	98
5.2	Influence de la fiabilité des prédictions du module DSS sur la qualité de la décision prise	100
5.3	Déroulement du processus d'expérimentation pour un espace de conception partiellement ordonné	102
5.4	Mise en évidence des limites du processus d'expérimentation présenté au chapitre 4 en cas d'erreurs de classement	104
5.5	Illustration de l'algorithme 3 après la détection d'une incohérence de type (a) ou (b)	107
5.6	Exemple de réorganisation de l'échelle de performance après une exploration offensive réussie	107
5.7	Algorithme de supervision pour un espace de conception partiellement ordonné .	111
5.8	Exemple de réorganisation de l'échelle de performance après une exploitation erronée des connaissances	113
5.9	Scénario de communication adopté pour l'étude de cas considérée	113
5.10	Espace de classification associé au nouveau scénario de reconfiguration	117
5.11	Pourcentage d'expériences de conception n'impliquant aucune mauvaise décision lors de la phase de recherche d'une configuration adaptée (CG_2)	118
5.12	Pourcentage d'expériences de conception se terminant par une décision optimale (CG_3)	119
5.13	Distance moyennée au comportement optimal en termes de rangs dans l'échelle d'optimalité (CG_4)	119
5.14	Nombre moyenné de décisions prises par expérience de conception (CG_5)	120

5.15	Nombre moyenné de mauvaises décisions prises par expérience de conception (CG_6)	121
5.16	Nombre moyenné d'incohérences identifiées par expérience de conception (CG_{12})	122
5.17	Évaluation de l'efficacité du processus d'expérimentation dans sa recherche d'un modèle prédictif fiable (CG_{13-16})	122
5.18	Analyse de la réponse du moteur cognitif aux problèmes de conception posés (CG_{17-23})	124
5.19	Taux d'exploration moyenné sur l'ensemble des automates d'apprentissage ($PSSE_1$)	125
5.20	Fiabilité des prédictions des modules DSS (FMP_4) et CAP (FMP_2) comparée à la qualité des décisions prises par le moteur cognitif (FMP_1)	126
5.21	Evolution de la carte d'adéquation avec l'expérience du moteur cognitif (FMP_3)	128
5.22	Evolution du nombre de règles à l'intérieur des bases de connaissances	129
5.23	Construction des graphes de robustesse à partir de connaissances expertes plus instructives sur l'espace de conception	132
5.24	Nombre moyenné de mauvaises décisions prises par expérience de conception en fonction de la qualité des connaissances expertes embarquées (CG_5)	133
5.25	Fiabilité des prédictions du module CAP en fonction de la qualité des connaissances expertes embarquées (FMP_2)	133
5.26	Nombre moyenné d'incohérences identifiées par expérience de conception en fonction de la qualité des connaissances expertes embarquées (CG_{12})	134
5.27	Evolution de la taille des bases de connaissances du moteur cognitif en fonction de la qualité des connaissances expertes embarquées (CMP_3)	134
5.28	Evolution de la carte de performance pour deux contraintes de performance différentes	135
5.29	Augmentation du nombre de descripteurs de l'environnement radio (apprentissage d'une carte à 4 dimensions)	136
6.1	Développement de systèmes intelligents basés sur des modules connexionnistes évolutifs (extrait de [146])	138
6.2	Organisation structurelle du moteur cognitif avancé	142
6.3	Organisation fonctionnelle du moteur cognitif	143
6.4	Décomposition de l'espace de conception en plusieurs sous-espaces de classification simplifiés	147
B.1	Application de la règle déterministe d'exploration offensive	161
B.2	Evolution de la carte d'exploration ($PSSE_2$) avec l'expérience du moteur cognitif pour la première étude de cas	163
B.3	Evolution de la carte d'exploration ($PSSE_2$) avec l'expérience du moteur cognitif pour la seconde étude de cas	164

Liste des tableaux

2.1	Spécificités de l'approche experte	22
2.2	Spécificités de l'approche exploratoire	24
2.3	Spécificités de l'approche prédictive	26
3.1	Politiques d'optimisation considérées (OP_{1-3})	37
3.2	Objectifs dépendants ou indépendants de l'environnement radio	39
3.3	Options de configuration considérées	49
3.4	Étude comparative des atouts/faiblesses de plusieurs techniques d'intelligence artificielle (d'après [128])	57
3.5	Matrice de décision multi-critères	62
4.1	Indicateurs de performance pour l'évaluation du moteur cognitif	79
4.2	Matrice de confusion pour le cas étudié	83
4.3	Comparaison des indicateurs de performance BRD_{1-3} pour des radios non flexibles et flexibles	94
4.4	Influence de la politique d'optimisation sur les indicateurs de performance	95
5.1	Indicateurs de performance supplémentaires pour l'évaluation du moteur cognitif généralisé	114
5.2	Configurations disponibles pour la conception du lien radio	116
5.3	Paramètres du moteur cognitif	117
5.4	Comparaison des indicateurs de performance BRD_{1-3} pour des radios non flexibles et flexibles	130
5.5	Influence des connaissances expertes (niveau 1, 2 ou 3) sur le comportement du moteur cognitif	131
7.1	Contributions des travaux de recherche réalisés dans le cadre de la thèse	156
A.1	Notations	159

Introduction

Sommaire

1.1	Contexte	1
1.1.1	Conception statistique d'un système radio	1
1.1.2	Vers une adaptation dynamique autonome	4
1.1.3	Concept et champs d'application de la radio cognitive	6
1.2	Objectifs de la thèse	7
1.3	Organisation du document	7

1.1 Contexte

1.1.1 Conception statistique d'un système radio

La conception d'un système de communication sans fil est un processus complexe guidée par une démarche logique schématisée à la figure 1.1.

La proposition d'un nouveau service (ou l'amélioration d'un service existant) résulte souvent d'un besoin mis en avant par la stratégie ou le marketing en fonction des progrès technologiques et de l'évolution du marché. L'expression du besoin reste encore fonctionnelle (*e.g.* pouvoir accéder à sa messagerie personnelle en situation de mobilité) ce qui laisse place à un grand nombre de solutions potentielles.

La phase d'analyse transforme la description fonctionnelle du besoin en une série de spécifications techniques guidant le concepteur dans sa recherche d'une solution adaptée. Les ingénieurs système s'accordent sur un ensemble d'indicateurs de performance (*e.g.* taux d'erreurs binaires, débit, latence) et définissent des valeurs cibles garantes du service demandé. Le système doit pouvoir maintenir cette qualité de service pour la majorité des situations rencontrées par le futur utilisateur. Or la fiabilité du lien radio varie grandement en fonction du temps et du lieu. Ce problème est géré traditionnellement en définissant un pire cas d'utilisation représentant les plus mauvaises conditions de propagation supportées par la norme. C'est pour ce pire cas que les contraintes de performance s'appliquent.

En plus de cet objectif d'efficacité, les solutions techniques développées doivent aussi respecter des contraintes de faisabilité ce qui limite encore plus le domaine d'action des concepteurs. Pour un système radio, il est nécessaire de tenir compte de deux facteurs. D'une part, la solution n'a d'intérêt que si elle peut être réalisée physiquement. Les concepteurs sont donc dépendants des progrès technologiques. D'autre part, le spectre radio-fréquence est une ressource limitée à

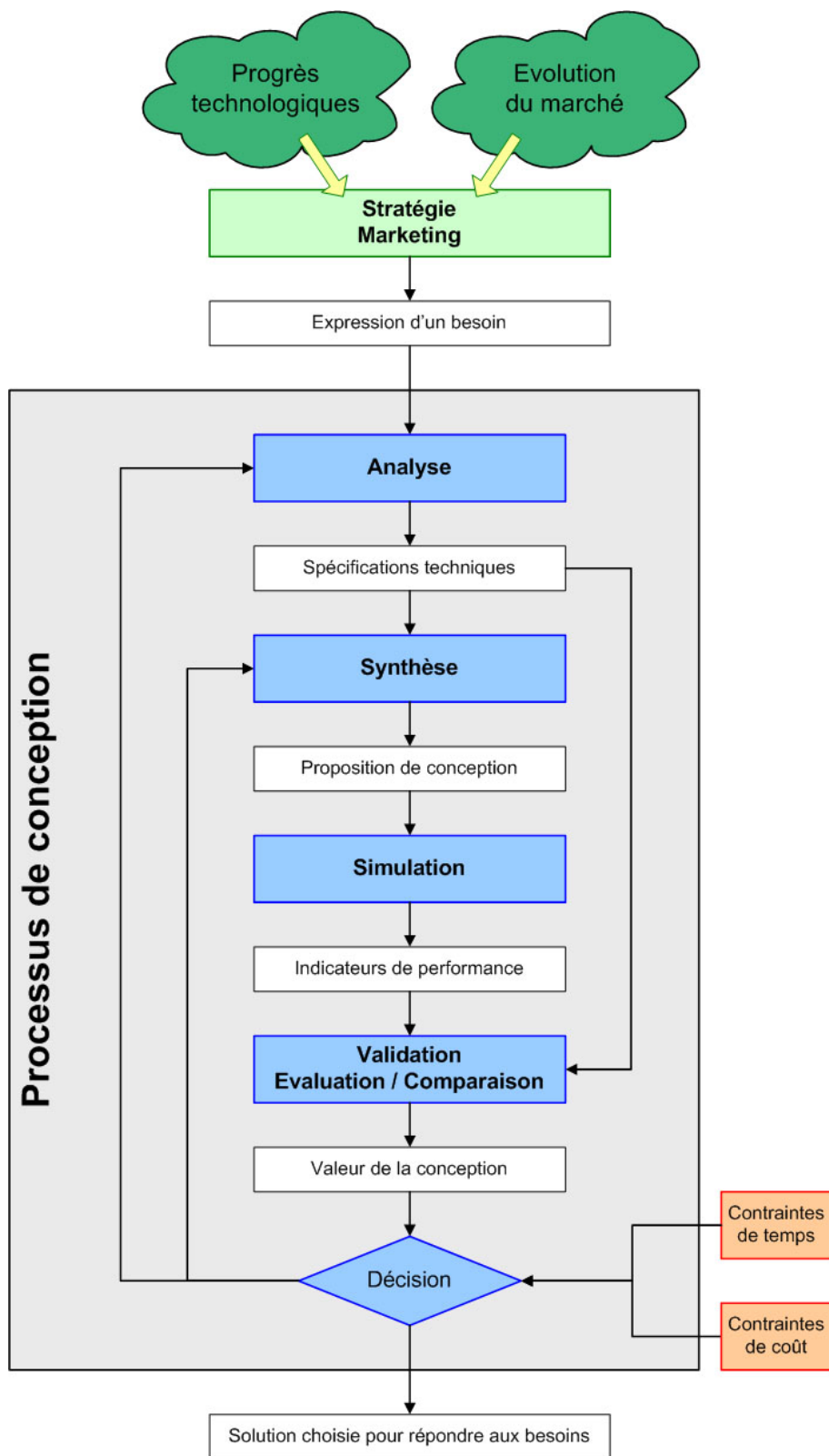


FIGURE 1.1 – Processus de conception d'un système radio

partager entre les services et les utilisateurs. Cette contrainte de coexistence justifie la mise en place d'un cadre de régulation pour contrôler les interférences.

Au-delà de ces spécifications techniques, le système doit être conçu dans un souci d'optimisation afin d'améliorer la qualité de service (*e.g.* réduire la consommation d'énergie des équipements portables) tout en maximisant les profits (*e.g.* diminuer les coûts opérationnels et matériels, augmenter la capacité du réseau).

Les concepteurs du système tiennent compte des spécifications pour générer une solution lors de la phase de synthèse. Le canal radio est un canal de communication difficile mais bien modélisé [1, 2]. La recherche en traitement du signal s'est concrétisée par de nombreuses techniques [3, 4] permettant d'exploiter au mieux les caractéristiques du canal (*e.g.* techniques d'accès multiples, techniques de diversité) et de compenser ses effets néfastes (*e.g.* codes correcteurs d'erreurs, égalisation). Le concepteur dispose donc d'un grand nombre de briques élémentaires qu'il peut assembler pour former des solutions prometteuses. Il s'appuie, pour cela, sur sa compréhension du canal radio, son expertise et son expérience pour estimer l'impact cumulé du canal et des blocs de traitement du signal sur les indicateurs de performance.

La solution envisagée est ensuite simulée, voire testée en conditions réelles afin de vérifier les choix du concepteur. La simulation s'appuie sur des modèles de canaux issus d'une campagne de mesures intensive [5]. Ces modèles sont voulus aussi fidèles que possible pour éviter le prototypage coûteux d'une solution non satisfaisante. Une simplification excessive du canal pourrait en effet donner lieu à des écarts de performance importants et induire le concepteur en erreur sur la viabilité de sa solution.

Lors de la phase de validation, le concepteur vérifie si sa solution est conforme aux spécifications. Le système conçu répond au problème de conception s'il respecte l'ensemble des contraintes imposées. La solution est éliminée dans le cas contraire et le concepteur retourne en phase de synthèse. Il peut aussi décider de revoir les spécifications à la baisse si elles s'avèrent trop contraignantes pour la mise en place d'une solution effective dans des délais acceptables. La phase de validation se poursuit par des activités d'évaluation et de comparaison qui justifient les choix successifs et témoignent des recherches d'optimisation du compromis global entre les différents objectifs. Les forces et les faiblesses des solutions acceptables sont mises en avant afin de converger progressivement vers une solution unique optimisée pour le service considéré compte tenu des contraintes de temps et de coût.

La méthode de conception classique a l'avantage de garantir des performances minimales pour le pire cas d'utilisation mais elle conduit à deux situations problématiques. Premièrement, le système radio se retrouve surdimensionné quand il évolue dans un canal plus favorable que le pire cas. Les objectifs ne sont donc pas autant optimisés qu'ils le devraient. Le système radio aurait pu utiliser une configuration alternative pour diminuer la consommation d'énergie, réduire l'empreinte spectrale du signal et/ou libérer plus rapidement le canal par augmentation du débit. Deuxièmement, le système radio ne peut plus délivrer le service au-delà du pire cas. Une configuration plus complexe aurait pu maintenir le service temporairement en attendant le retour à des conditions de propagation plus favorables.

La flexibilité croissante des équipements de communication change la donne et il est désormais possible de songer à optimiser certains compromis en fonction de l'état du canal radio. La radio autrefois figée devient adaptative.

1.1.2 Vers une adaptation dynamique autonome

Une radio consomme des ressources lorsqu'elle communique. Certaines ressources sont personnelles (*e.g.* énergie) alors que d'autres sont partagées entre tous les utilisateurs (*e.g.* spectre radio). Les objectifs/contraintes de conception ne cessent d'augmenter avec la popularité croissante des communications sans fils et les exigences toujours plus grandes de qualité de service (QoS : Quality of Service). Il est donc nécessaire d'assurer une gestion toujours plus efficace des ressources disponibles. Cet objectif d'optimisation peut être atteint en suivant deux approches différentes mais complémentaires.

La première approche consiste à développer de nouvelles techniques pour transmettre l'information plus rapidement avec une plus grande fiabilité (*e.g.* modulations avancées, principe turbo, traitements multi-antennes). Cette approche a longtemps été privilégiée.

La seconde approche passe par le développement d'équipements plus flexibles. Les concepteurs du système disposent alors d'un certain nombre de paramètres que l'équipement peut modifier en ligne pour s'adapter à son environnement. Les variations de l'environnement ne sont donc plus considérées comme un handicap mais comme une opportunité d'optimiser l'usage des ressources en fonction de la difficulté de la situation. La radio peut ainsi profiter de conditions favorables pour diminuer sa consommation d'énergie, réduire son empreinte spectrale (et ainsi augmenter la capacité du réseau) et/ou améliorer la QoS délivrée à l'utilisateur. Cette approche est susceptible de limiter énormément le gaspillage de ressource. Son intérêt n'est donc pas nouveau.

Les systèmes de communication de seconde génération (*e.g.* GSM) peuvent déjà adapter leurs opérations aux conditions de propagation. Le contrôle de puissance à l'émission permet de garantir une fiabilité minimum du lien radio tout en réduisant les interférences générées. En réception, les performances du système sont optimisées à l'aide d'algorithmes adaptatifs capables d'ajuster leur complexité en fonction des variations du canal (*e.g.* égaliseur MLSE pour la norme GSM). L'adaptation reste cependant encore limitée.

L'adaptation passe à un niveau supérieur avec l'arrivée d'architectures matérielles plus flexibles. Pour atteindre des débits toujours plus élevés, les radios peuvent désormais ajuster leur schéma de modulation et de codage en fonction des informations remontées par le récepteur sur la qualité du canal (*e.g.* 802.11, GPRS, EDGE, HSDPA). Les réseaux de troisième génération gagnent aussi en flexibilité et peuvent allouer dynamiquement les ressources en fonction de leur charge. La norme HSDPA permet ainsi d'affecter jusqu'à 15 codes à un utilisateur afin d'augmenter le débit de la communication.

Les propriétés d'adaptation ne cessent de s'améliorer avec les nouveaux standards de communication mais le lien radio reste encore loin d'être complètement optimisé.

Les mécanismes d'adaptation actuels sont définis sur des périmètres d'action étroits (accès au spectre contraint, flexibilité limitée des équipements, standards définis pour des tâches spécifiques). La radio réagit suivant une logique simple en fonction d'un nombre limité d'événements anticipés. L'algorithme de décision prend généralement la forme d'une suite d'expressions conditionnelles de type IF-THEN-ELSE [6]. Les règles utilisées s'appuient sur une table de correspondance associant une action prédéterminée à un ensemble de valeurs de l'espace d'entrée (comportement déterministe). Le schéma de modulation et de codage appliqué varie par exemple avec la valeur du rapport signal sur interférences plus bruit (SINR : Signal to Interference-plus-Noise Ratio) estimé. Les valeurs seuils sont déterminées lors de la phase de conception du système à partir de simulations intensives. La table n'est plus susceptible d'évoluer en sortie de production.

Ces mécanismes d'adaptation rapide peuvent être vus comme une réponse réflexe de la radio à

un ensemble de stimuli prédéfinis [7,8]. Il est cependant nécessaire de pousser l'adaptation encore plus loin pour que la radio puisse exploiter les variations de son contexte et ainsi optimiser ses performances avec une meilleure gestion des ressources. Cet impératif se justifie sur les quatre plans suivants :

- **Conjoncture technologique favorable** : Les progrès technologiques permettent désormais d'envisager le remplacement des circuits dédiés par des processeurs capables de traiter l'information en temps réel pour une consommation raisonnable. Cette évolution donne vie au concept de radio logicielle introduit en 1992 par Mitola [9]. La radio logicielle idéale promet une flexibilité et une agilité sans limite [10, 11]. Toutes ses fonctionnalités sont codées sous la forme de programmes dynamiquement chargés/modifiés en fonction du signal à traiter. La radio supporte plusieurs standards de communication (convergence des services) et elle peut moduler/démoduler une multitude de formes d'onde afin de transmettre l'information le plus efficacement possible. Elle est aussi capable d'évoluer rapidement en téléchargeant des modifications logicielles (correctifs de sécurité, intégration de nouvelles capacités, mise à jour d'informations). La radio purement logicielle n'existe pas encore aujourd'hui suite aux limitations technologiques. De fortes contraintes existent sur les architectures matérielles, notamment autour du front end radio (*e.g.* conversion analogique/numérique). Les radios dépendent donc encore de circuits matériels spécialisés plus ou moins paramétrables (ASICs, FPGAs) mais cette tendance s'affaiblit avec le temps au fur et à mesure que le traitement logiciel se rapproche de l'antenne. Cette flexibilité nouvelle ouvre de nombreuses perspectives sur les possibilités d'adaptation de la radio.
- **Problème d'échelle** : Le succès des communications sans fil s'est considérablement amplifié ces dernières années. Le taux de pénétration des équipements mobiles ne cesse d'augmenter et les ventes d'ordinateurs portables commencent à dépasser celles des stations de travail. Les applications M2M (Machine to Machine) se multiplient pour former des réseaux toujours plus nombreux, plus denses et plus intelligents (Internet of Things [12]). Cette explosion du nombre d'objets communicants risque de créer de sérieux problèmes de coexistence sans une refonte majeure des mécanismes d'accès au spectre radio. Les mécanismes actuels sont contraints par un système de régulation ultra protecteur reposant sur l'attribution statique de bandes fréquentielles pour chaque service/opérateur. Ce système est inefficace comme l'attestent de nombreuses campagnes de mesures. Certaines bandes fréquentielles se retrouvent proches de la saturation alors que d'autres ne sont pas utilisées ou de manière irrégulière en fonction du temps et du lieu [13]. L'accès opportuniste au spectre apparaît donc comme nécessaire pour homogénéiser l'utilisation du spectre et permettre la coexistence de cette multitude d'objets communicants. Le partage dynamique du spectre s'accompagne néanmoins de nombreux défis techniques et réglementaires nécessitant une intelligence plus évoluée de la part de la radio.
- **Problème de complexité** : Autrefois simple et peu encombré, le paysage radio est devenu progressivement hétérogène et très complexe (prolifération des standards radios, diversité des services proposés). Le nombre d'interactions a considérablement augmenté (problème d'échelle) et les possibilités de reconfiguration sont toujours plus nombreuses avec l'arrivée de la radio définie logicielle. La prise de décision ne peut donc plus se faire aussi simplement qu'auparavant si l'on veut réellement optimiser l'usage des ressources. Des mécanismes de raisonnement plus génériques et plus puissants sont nécessaires pour permettre à la radio (a) de s'adapter à la grande diversité des contextes rencontrés, (b) de tenir compte de plusieurs objectifs souvent conflictuels et (c) de sélectionner l'opportunité la plus pertinente parmi une multitude de choix disponibles. Les concepteurs de systèmes radio gagneraient

également à introduire des capacités d'apprentissage dans les équipements afin d'augmenter leur autonomie décisionnelle face à des situations inattendues. L'apprentissage permettrait, de plus, d'automatiser la phase d'acquisition des connaissances donc de réduire le temps de mise sur le marché de nouvelles technologies. Il permettrait, enfin, d'assurer une optimisation plus poussée du comportement de la radio.

- **Conjoncture économique ultra compétitive** : La course à l'innovation s'est accélérée ces dernières années avec la convergence des industries des télécommunications et de l'informatique. Par exemple, Microsoft s'intéresse désormais à des thématiques de recherche hors de son champ d'expertise traditionnel comme l'accès opportuniste au spectre [14] ; France Telecom a modifié progressivement sa stratégie pour s'affirmer comme un fournisseur de services internet en plus de son rôle historique d'opérateur. Dans cet environnement fortement concurrentiel, il est nécessaire de séduire l'utilisateur en lui fournissant des services personnalisés, optimisés et simples. Ces objectifs économiques ne pourront être atteints sans le développement d'une couche intelligente embarquée agissant dans le meilleur intérêt de l'utilisateur tout en lui cachant la complexité des mécanismes sous-jacents [15].

Ces quatre raisons montrent que la radio ne peut plus se contenter d'être adaptative. Il faut désormais la rendre cognitive.

1.1.3 Concept et champs d'application de la radio cognitive

La radio cognitive est un concept technologique poussant vers une adaptation totale et autonome des équipements vis-à-vis de leur contexte opérationnel. Une radio cognitive s'appuie sur une observation intelligente de son environnement pour adopter la meilleure ligne de conduite en toutes circonstances à l'aide d'actions réactives et proactives. Elle prend ses décisions dans le meilleur intérêt de l'utilisateur et du réseau et elle apprend de ses expériences pour toujours améliorer ses performances. La radio passe ainsi d'un simple exécuter aveugle de protocoles prédéfinis à un agent intelligent, sensible au domaine radio et autonome.

La radio cognitive a attiré une grande attention depuis son introduction en 1999 par Mitola [16]. La vision de Mitola projette le concept loin dans l'avenir en considérant le terminal radio comme un assistant personnel capable d'anticiper les besoins de l'utilisateur pour y répondre de la meilleure façon. Cette vision futuriste a depuis laissé place à des visions plus technologiques permettant une meilleure organisation des efforts de recherche sur le sujet. Nous noterons cependant qu'il aura fallu un certain temps à la communauté pour se mettre d'accord sur la définition et les propriétés attendues d'une radio cognitive [17, 18, 19, 20]. Encore maintenant, il est préférable de raisonner en termes de degrés de cognition pour distinguer les solutions proposées en fonction de leurs capacités intrinsèques [17, 21, 8].

La radio cognitive reste un sujet de recherche très vaste, stratégique pour tout acteur travaillant dans le domaine des télécommunications. Le concept de radio intelligente a donné naissance à un grand nombre d'applications particulièrement prometteuses [22] comme par exemple :

- *L'accès opportuniste au spectre* : Maximiser l'efficacité spectrale par une allocation dynamique des fréquences et une gestion efficace des interférences (reçues et émises) dans le respect du cadre de régulation.
- *La conception cognitive du lien radio* : Optimiser les performances du système radio en sélectionnant les blocs de traitement du signal les plus appropriés.
- *La sélection de la meilleure technologie d'accès radio* : Identifier le réseau d'accès répondant au mieux aux besoins de l'utilisateur pour un coût aussi faible que possible.

- *L'adaptation de la topologie d'un réseau* : Augmenter la couverture et la capacité d'un réseau par une coopération entre les radios.

Étant donné son champ d'application, la radio cognitive est un sujet fortement multidisciplinaire exigeant des compétences en traitement du signal, réseau, analyse statistique, ingénierie logicielle et matérielle, prise de décision autonome, et apprentissage artificiel pour en citer quelques unes. Elle demande aussi de revisiter la régulation du spectre [23] ainsi que les modèles économiques développés pour les communications sans fils [24].

Nous revenons amplement sur le concept de radio cognitive dans le reste du document. Le lecteur intéressé peut aussi se référer à des textes de portée plus générale comme [25, 26, 27, 28] .

1.2 Objectifs de la thèse

Avec la complexité croissante des problèmes d'ingénierie, il est nécessaire de concevoir des systèmes de plus en plus intelligents, capables de reproduire certaines facultés de l'esprit humain en toute autonomie (*e.g.* reconnaissance, analyse, raisonnement, apprentissage). Le domaine des télécommunications n'échappe pas à cette tendance comme le montrent l'évolution des technologies radio et la montée en puissance de la radio cognitive.

La radio cognitive pose une problématique très large à l'interface entre deux domaines scientifiques : les télécommunications et l'intelligence artificielle. Les capacités de raisonnement et d'apprentissage sont tributaires des techniques d'intelligence artificielle introduites dans l'équipement radio. La modélisation de l'environnement et les connaissances expertes sont directement issues du monde des télécommunications.

Cette thèse est consacrée à l'intelligence embarquée. L'objectif final est de formaliser un processus cognitif capable de superviser la reconfiguration dynamique des équipements radio. Dans ce document, nous détaillons l'architecture d'un moteur cognitif largement inspiré du processus de conception illustré à la figure 1.1. Le mimétisme avec l'expert humain est réalisé aux travers de techniques d'intelligence artificielle couplées à une modélisation efficace du problème de conception à la volée. Le moteur cognitif proposé guide l'adaptation de la radio à son environnement tout en apprenant de ses expériences pour améliorer ses performances.

1.3 Organisation du document

Le chapitre 2 développe le concept de radio cognitive en mettant l'accent sur l'intelligence embarquée. Les approches cognitives existantes sont analysées afin d'en dégager leurs forces et leurs faiblesses. Nous partons ensuite de cette étude critique pour justifier l'approche que nous avons suivie dans nos travaux.

Le chapitre 3 présente le scénario de reconfiguration dynamique adopté, expose les réflexions à l'origine du moteur cognitif proposé et justifie les choix méthodologiques réalisés.

Les deux chapitres suivants détaillent l'organisation structurelle et fonctionnelle du moteur cognitif avant d'analyser son comportement sur un cas de conception cognitive du lien radio. Les mécanismes cognitifs décrits au chapitre 4 sont valides pour un espace de conception totalement ordonné. Le moteur cognitif peut alors profiter de relations d'ordre fiables entre les alternatives pour structurer son exploration de l'espace de conception. Les connaissances embarquées sont rarement aussi instructives pour des études de cas plus complexes. Les mécanismes cognitifs sont donc généralisés au chapitre 5 afin de s'affranchir de cette contrainte par l'apprentissage.

Le chapitre 6 propose des réflexions pour optimiser les mécanismes cognitifs et les rendre opérationnels sur des problèmes concrets de reconfiguration dynamique tels qu'ils seront posés par les futures radios logicielles.

Le chapitre 7 dresse le bilan des recherches effectuées avant de synthétiser les contributions marquantes des travaux exposés.

L'annexe A décrit la modélisation analytique du système de communication reconfigurable supervisée par le moteur cognitif simulé.

L'annexe B donne des compléments d'information sur les mécanismes cognitifs proposés (exemples illustratifs et résultats supplémentaires).

Intelligence artificielle appliquée aux télécommunications

Sommaire

2.1	Introduction d'un cycle cognitif dans une radio flexible	10
2.2	Conception d'un système de communication cognitif	11
2.2.1	Contexte opérationnel d'une radio	11
2.2.2	Organisation interne d'une radio cognitive	13
2.2.3	Axes de recherche	13
2.3	Conception d'un moteur cognitif	16
2.3.1	Détermination du problème de conception	17
2.3.2	Résolution du problème de conception	18
2.3.3	Synoptique d'un moteur cognitif	19
2.4	Approches cognitives existantes	19
2.4.1	Approche experte	20
2.4.2	Approche exploratoire	22
2.4.3	Approche prédictive	24
2.5	Approche adoptée dans nos travaux	26

Une radio cognitive est un agent intelligent capable de s'adapter à son contexte opérationnel suivant des objectifs de haut-niveau. Ces objectifs consistent à :

- Respecter le cadre de régulation contrôlant l'accès au spectre,
- Satisfaire les besoins de l'utilisateur en termes de qualité de service,
- Assurer une gestion optimisée des ressources disponibles (radios, réseaux et matérielles).

Ce niveau d'adaptation ne peut être atteint sans l'introduction d'une couche intelligente capable de contrôler la reconfiguration de l'équipement face aux situations dynamiques et imprévisibles rencontrées. Le monde des télécommunications a donc besoin d'évoluer et de profiter des efforts de recherche en intelligence artificielle pour augmenter l'autonomie décisionnelle des radios.

Dans ce chapitre, nous présentons des éléments de réflexion guidant la conception d'une radio cognitive et de sa couche intelligente plus particulièrement. Nous faisons ensuite un bref tour d'horizon des approches cognitives existantes et nous positionnons nos travaux par rapport à l'état de l'art.

2.1 Introduction d'un cycle cognitif dans une radio flexible

Une radio cognitive est un système radio reconfigurable se comportant de manière intelligente et autonome suite à l'introduction d'un *cycle cognitif* lui permettant d'analyser son contexte et d'agir en conséquence.

De nombreuses variantes ont été proposées [16,29,30,31] mais nous pouvons toutes les ramener au cycle générique présenté à la figure 2.1. Ce cycle fait intervenir trois capacités essentielles à tout agent autonome [32] :

- La capacité d'*observer* : La radio dispose de capteurs pour récupérer des informations sur son contexte opérationnel.
- La capacité d'*agir* : La radio influe sur son environnement au travers de ses actionneurs. Le signal émis occupe des ressources spectrales et les traitements appliqués consomment des ressources énergétiques. La radio peut se reconfigurer et ainsi moduler cette influence en fonction de la situation et des objectifs de QoS.
- La capacité de *réfléchir* : La radio exploite les informations récupérées pour identifier la configuration la plus adaptée et apprendre de ses expériences.

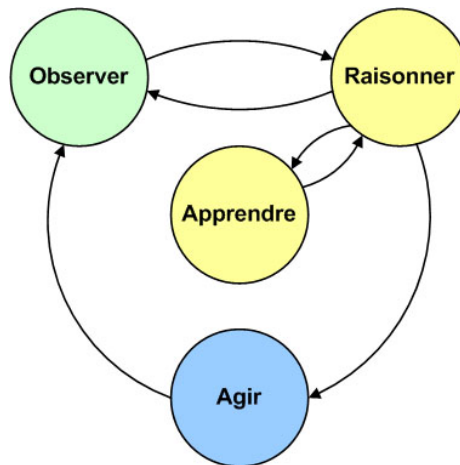


FIGURE 2.1 – Cycle cognitif générique

L'activité cognitive peut être décomposée en deux processus complémentaires : le raisonnement et l'apprentissage [33,18]. Le raisonnement est un processus logique qui permet d'aboutir à une conclusion à partir de propositions et d'observations. Une radio cognitive peut recourir à deux niveaux de raisonnement. Le premier niveau garantit l'autonomie décisionnelle de la radio. La radio dispose de connaissances qu'elle exploite pour identifier une action ou une stratégie en accord avec ses observations. Avec ce simple niveau de raisonnement, la radio est fortement dépendante de la qualité et de la précision des connaissances embarquées. Le deuxième niveau de raisonnement permet de diminuer cette dépendance. La radio s'appuie sur une démarche scientifique pour analyser son propre comportement dans une optique de perfectionnement par apprentissage (Haykin parle de "methodology of understanding-by-building" [29]). Un élément de créativité est alors nécessaire pour que la radio propose de nouvelles actions la poussant vers une meilleure compréhension de son contexte [34]. L'apprentissage permet ensuite d'acquérir et/ou d'actualiser les connaissances manipulées par les processus de raisonnement de premier niveau.

Ce cycle générique laisse libre cours à de nombreuses interprétations comme nous le verrons à la section 2.4. Le second niveau de raisonnement n'est pas toujours jugé nécessaire, notamment

pour l'accès opportuniste au spectre [17,21].¹ L'apprentissage n'est alors plus possible en ligne et la radio ne peut progresser qu'à l'aide d'opérateurs humains diffusant de nouvelles connaissances obtenues à partir de simulations hors-ligne [7]. Cette situation d'apprentissage fait référence au mode "prière" mis en avant par Mitola [35]. Ce mode peut aussi être qualifié de mode assisté.

L'apprentissage en ligne fait intervenir des algorithmes souvent jugés trop coûteux en ressources de calcul pour être appliqués lors de la phase opérationnelle. Mitola conseille en effet de profiter des périodes de veille de la radio pour lancer les mécanismes d'apprentissage sans nuire au confort de l'utilisateur [35].² La radio ajuste alors ses connaissances en fonction d'un lot d'informations récupérées et mémorisées lors de la phase opérationnelle (mode batch).

Au-delà de ces considérations sur l'apprentissage, chaque solution se distingue aussi par sa manière de représenter les connaissances et de les exploiter (premier niveau de raisonnement). Nous ferons état de ces différences dans la section 2.4 et nous dégagerons les avantages et les axes d'amélioration des approches cognitives existantes. Cette discussion conduira à la présentation et la justification de notre approche à la section 2.5.

2.2 Conception d'un système de communication cognitif

La figure 2.2 représente une architecture haut-niveau pour un système de communication cognitif interagissant avec son contexte opérationnel. Nous commentons la figure en deux temps. Nous revenons tout d'abord sur la notion de contexte opérationnel pour une radio. Nous détaillons ensuite l'organisation interne du système cognitif. Nous terminons enfin cette section par une présentation des axes de recherche principaux à explorer avant d'atteindre un système réellement effectif.

2.2.1 Contexte opérationnel d'une radio

Le contexte opérationnel regroupe l'ensemble des conditions sous lesquelles la radio opère. Il peut être analysé selon cinq dimensions :

- *La dimension du régulateur* : L'accès au spectre radio est contrôlé par un ensemble de lois définies par des organismes de régulation. Ces lois dictent les règles de bonne conduite qu'il est obligatoire de respecter pour utiliser le spectre. En fonction du temps et du lieu, elles interdisent certaines régions spectrales réservées à des utilisateurs prioritaires comme les équipements d'intervention d'urgence (policiers, pompiers, médecins) ou les radios traditionnelles opérant sur des bandes fréquentielles achetées et contrôlées par un fournisseur de service. De plus, ces lois assurent un comportement policé des radios opportunistes qui ne peuvent transmettre sur une bande fréquentielle si elle est détectée comme déjà occupée pour une autre communication. Elles contingentent aussi la quantité d'interférences que la radio peut s'autoriser à générer dans et en dehors de la bande sélectionnée/négociée. De part son aspect réglementaire, cette dimension impose des contraintes strictes sur la conception du lien radio notamment en termes de puissance maximale d'émission et de bande passante atteignable.

1. Une radio opportuniste peut se contenter d'adopter une politique prédéfinie déclenchée en fonction des mesures sur le spectre radio. Nous ne partageons pas cette idée couramment admise. Nous pensons en effet que l'apprentissage reste un élément important dans une optique d'optimisation des performances.

2. Nos travaux vont au-delà de cette recommandation puisque nous avons mis en place une méthode simple et efficace d'acquisition incrémentale des connaissances.

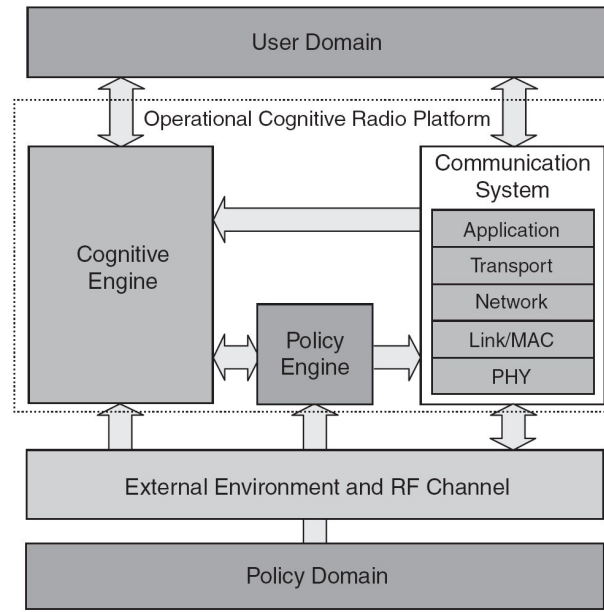


FIGURE 2.2 – Architecture haut-niveau d'une radio cognitive (extrait de [25])

- *La dimension de l'opérateur* : L'opérateur doit respecter rigoureusement le cadre de régulation en vigueur mais il peut à son tour imposer des contraintes plus ou moins souples afin de répartir correctement les ressources spectrales entre ses abonnés. Il cherche avant tout à maximiser la capacité de son réseau afin d'accueillir un nombre toujours plus grand d'utilisateurs dans les meilleures conditions possibles. Cette dimension va donc influencer la phase de conception en orientant la radio vers des configurations plus ou moins gourmandes en ressources spectrales en fonction de la charge du réseau.
- *La dimension de l'utilisateur* : L'utilisateur est directement concerné par la qualité du service fourni par le lien radio. La qualité d'un service est une notion qualitative prenant un sens quantitatif différent pour chaque application. Les applications temps-réel valorisent les communications de faible latence alors que l'efficacité du lien radio se mesure en termes de débit pour les applications non temps-réel. Étant donné que l'utilisateur attend la meilleure QoS possible, cette dimension impose des objectifs de performance pour la radio.
- *La dimension de l'équipement* : La radio a un périmètre d'action délimité par la flexibilité et l'agilité de sa plateforme matérielle. Elle ne peut émettre au-delà de la puissance maximale de son amplificateur et elle ne dispose que d'un nombre limité de traitements à appliquer. Cette dimension limite donc les capacités d'adaptation de la radio. Au-delà de ses possibilités de reconfiguration, une radio cognitive doit aussi être sensible à sa consommation d'énergie (et au temps de batterie restant), à sa puissance de calcul et à ses capacités mémoire. Cette dimension va alors infléchir sur le processus de conception en favorisant les configurations qui maximisent l'autonomie de la batterie (par exemple).
- *La dimension associée à l'environnement radio* : L'environnement radio fournit le support de communication utilisé par le système. Son influence sur les performances de la communication dépend de la forme d'onde développée et du récepteur mis en place. Le lien radio gagnera en efficacité si le signal est modulé sur une large bande passante à moins qu'il ne soit perturbé par une ou plusieurs sources d'interférence. De la même manière, la QoS sera fortement dégradée si la radio évolue dans un canal très sélectif sans une structure

complexe de récepteur. Un récepteur plus simple serait par contre envisageable en présence d'un trajet direct. Cette dimension permet donc d'établir les frontières de décision entre les alternatives disponibles [19].

2.2.2 Organisation interne d'une radio cognitive

Le comportement intelligent d'une radio est dû à l'introduction d'un moteur cognitif structurant les opérations de modélisation, de prise de décision et d'apprentissage nécessaires au contrôle dynamique du processus de reconfiguration. Le système de communication alimente le moteur cognitif avec les informations qu'il récupère sur son contexte opérationnel (*e.g.* mesures sur l'environnement radio, indicateurs de performance, politiques diffusées, niveau de batterie restant). Le moteur cognitif analyse les exigences de l'utilisateur et de l'opérateur, les régimes de régulation ainsi que le lien physique afin d'identifier les objectifs de conception et les contraintes applicables. Il se base ensuite sur cette compréhension du contexte pour adapter les paramètres de la pile protocolaire du système de communication.

Le cadre de régulation est défini à l'aide de politiques mémorisant les contraintes d'accès au spectre dans un langage compréhensible par la machine. Les politiques sont diffusées localement et les contraintes peuvent évoluer en fonction du temps et du lieu. La radio doit respecter en permanence le cadre de régulation. Elle dispose pour cela d'un moteur d'inférence (*policy engine*) capable de bloquer toute solution violant les règles de régulation locales. Le moteur d'inférence est un composant cognitif appliquant un raisonnement déductif à partir d'informations exprimées dans un langage machine standardisé. Les capacités d'inférence du moteur peuvent être mises à profit pour d'autres activités. Par exemple, le moteur cognitif peut faire appel à lui pour analyser les possibilités de reconfiguration des objets communicants, pour identifier les objectifs opérationnels, voire pour proposer une stratégie de communication adaptée. Le moteur d'inférence sera exploité plus ou moins intensément en fonction de l'approche cognitive adoptée (*cf.* 2.4). Ce composant est néanmoins obligatoire ne serait-ce que pour garantir la légalité des propositions du moteur cognitif vis-à-vis du cadre de régulation.

2.2.3 Axes de recherche

La radio cognitive est un sujet de recherche particulièrement stimulant pour l'industrie des télécommunications. Nous avons identifié six axes majeurs en rapport avec le cadre de nos travaux.

Développement d'une plateforme cognitive reconfigurable

Dans la continuité des efforts de recherche en radio logicielle [10, 11], de nombreux travaux ont été réalisés sur la plateforme matérielle hébergeant la radio cognitive. Le traitement logiciel se rapproche progressivement de l'antenne même s'il reste encore de nombreux verrous technologiques à débloquer, notamment autour du front end radio [36]. Les plateformes proposées se montrent de plus en plus flexibles et agiles [37] et des projets de développement open-source apparaissent afin de faciliter la construction logicielle d'une radio. Nous pouvons citer notamment le projet GNU Radio [38, 39] proposé par la Free Software Foundation et le projet OSSIE [40, 41] développé principalement à Virginia Tech. Ces plateformes sont désormais conçues pour embarquer des fonctionnalités cognitives [39, 42, 43, 44, 45, 19]. Elles peuvent accueillir un moteur cognitif et dialoguer avec lui au travers d'une interface standardisée favorisant les échanges entre les deux entités (mise à disposition des informations contextuelles, implémentation des ordres de reconfiguration).

Caractérisation en ligne de l'environnement radio

La radio cognitive est vue comme une technologie particulièrement efficace pour améliorer l'efficacité spectrale. Pour accéder de manière opportuniste au spectre, une radio doit d'abord observer son environnement radio (phase de sondage ou sensing phase). Cette phase est la base de toute prise de décision. Elle doit donc être réalisée avec précision afin de minimiser les interférences lors de l'allocation dynamique des fréquences. De nombreuses techniques ont été proposées pour permettre au terminal d'analyser l'activité spectrale autour de lui [46,47,48,49,50,51]. L'environnement radio est difficile à observer de par son aspect hautement dynamique (variabilité dans le temps et l'espace) et sa nature statistique. Il est donc préférable de récolter un grand nombre d'informations pour augmenter la fiabilité du modèle et ainsi assurer une prise de décision pertinente. Le sondage collaboratif a été proposé comme une méthode efficace pour améliorer la qualité de la modélisation par rapport à une caractérisation individuelle [52,53]. Les informations prises par plusieurs radios collaboratives sont alors partagées et échangées afin d'aboutir à un modèle plus complet (caractérisation spatio-temporelle) et plus fiable (récupération de plusieurs réalisations d'une même variable aléatoire) de l'environnement.

Intelligence embarquée

L'intelligence embarquée pose un véritable défi pour la recherche car elle est la base de toute activité cognitive. La radio doit savoir modéliser efficacement son contexte opérationnel, identifier l'adéquation de sa configuration actuelle et proposer une nouvelle configuration à la volée dès qu'elle a une opportunité d'améliorer ses performances. L'expertise encodée dans l'équipement doit couvrir un large panel de situations afin d'assurer une fiabilité uniforme des prises de décision. Pour devenir pleinement autonome, la radio se doit d'intégrer des capacités d'apprentissage afin de pouvoir "improviser" face à une situation nouvelle. L'apprentissage permet aussi à la radio de se rapprocher progressivement d'un comportement optimal au fur et à mesure des expériences accumulées. Les mécanismes proposés doivent enfin permettre une prise de décision en temps réel sans introduire une complexité excessive pouvant nuire à l'autonomie énergétique de l'équipement. L'ensemble des exigences présentées forme le point de départ des travaux réalisés lors de cette thèse. L'analyse des solutions existantes sera détaillé à la section 2.4.

Organisation des connaissances à travers un langage standardisé

La mise au point de radios cognitives passe par le besoin d'organiser les connaissances sur le domaine radio en un langage structuré capable d'être traité en temps réel. Ce langage est nécessaire, ne serait-ce que pour définir les politiques de régulation réglementant l'accès au spectre. Il doit aussi permettre de représenter les informations sur les architectures matérielles, les modules logiciels, l'environnement radio ainsi que les services disponibles et leurs besoins. Cet axe de recherche est très fortement lié à l'approche experte exposée en 2.4.1.

Aspects protocolaires et problèmes de sécurité

Cet axe de recherche concerne les aspects protocolaires et les questions de sécurité soulevées par ce nouveau type d'équipements. Les radios vont échanger beaucoup plus d'informations qu'elles ne le faisaient auparavant. De nouveaux protocoles sont donc nécessaires pour le sondage collaboratif de l'environnement radio [54], pour le partage et la négociation du spectre [55,56], pour transférer les ordres de reconfiguration entre les équipements communicants et pour diffuser les politiques faisant respecter le cadre de régulation local. Le projet E2R [57] propose de mettre en place un canal pilote cognitif pour propager des informations facilitant le processus de

connexion des radios [58]. Moy [59] considère la technologie Ultra Large Bande comme particulièrement adaptée à cette tâche.

Avec le partage des informations, des problèmes de sécurité apparaissent. Quelles informations s'autorise-t-on à propager ? Comment garantir l'authenticité et la fiabilité des informations reçues ? Comment s'assurer du comportement des radios notamment vis-à-vis de la régulation [60] ? Comment se protéger face aux attaques malicieuses dérégulant les activités cognitives de la radio [61, 62] ? Autant de questions auxquelles il faudra répondre avant de pouvoir garantir un système cognitif fiable.

Réseau cognitif

Les systèmes de télécommunications radio cellulaires disposent aujourd'hui d'une infrastructure centralisée interagissant avec une multitude de terminaux qui lui sont attachés. Dans d'autres contextes, les équipements sont capables de s'affranchir d'une infrastructure de support en opérant en mode ad-hoc. Les systèmes formés sont d'une très grande complexité dans les deux cas car chaque élément du système doit assurer ses propres objectifs à partir d'un ensemble de ressources limitées et partagées. Une radio cognitive peut moduler son impact sur les autres utilisateurs dans l'intérêt global du réseau (*e.g.* réduire son empreinte spectrale). Ce résultat n'est pas garanti à première vue. Il est donc nécessaire d'analyser les interactions entre les nœuds du réseau afin de garantir un comportement global optimal caractérisé par une bonne QoS pour toutes les transmissions, un faible coût pour l'utilisateur et un grand profit pour l'opérateur (avec une diminution des coûts d'exploitation et d'entretien notamment). La théorie des jeux est un outil particulièrement adapté à cet exercice [29, 17, 63]. Elle permet d'étudier la prise de décision dans un environnement où plusieurs joueurs (*i.e.* les radios) interagissent. Le choix d'un comportement optimal est alors influencé par l'environnement radio mais aussi par les décisions des autres utilisateurs. Le problème du partage équitable des ressources peut aussi être traité dans le cadre des travaux sur les systèmes multi-agents [64, 65, 66]. Un système multi-agents se base sur la coopération, la coordination ou même la compétition entre les agents pour développer une intelligence collective se manifestant par un comportement complexe capable d'auto-organisation [67].

Lors du développement d'un réseau cognitif, il sera nécessaire de déterminer *qui* prend *quelle* décision. L'infrastructure de support sera très souvent favorisée étant donné qu'elle a des contraintes énergétiques moins fortes que les terminaux et qu'elle dispose d'une vision globale et complète du réseau. La centralisation des informations passe néanmoins par des efforts de signalisation importants diminuant l'efficacité spectrale et la réactivité du système. De plus, les radios n'ont pas toujours accès à une infrastructure. Les militaires sont amenés, par exemple, à évoluer dans un environnement adverse dont ils ne peuvent dépendre. De même, les situations de crise et d'urgence impliquent généralement une destruction au moins partielle des infrastructures. Dans ce cas, la décision pourrait être prise par une seule radio qui se verrait attribuer le rôle de "maître" par les autres radios devenues "esclaves". La décision pourrait aussi être distribuée entre les équipements afin d'accélérer la phase de décision tout en réduisant la consommation individuelle des ressources énergétiques [68].

La question du partage de l'intelligence a donc une grande importance et de nombreux schémas peuvent être proposés. La réponse à cette question dépendra des avancées techniques sur les mécanismes de prise de décision mais aussi de considérations économiques, les opérateurs ayant tendance à vouloir garder un certain contrôle sur leur base de clients.

Cette thèse est consacrée au problème de l'intelligence embarquée. Étant donné la difficulté de la problématique, nous avons décidé de faire quelques hypothèses simplificatrices quant aux autres axes de recherche :

- Toutes les informations sur le contexte de communication (*e.g.* état du canal, état énergétique de l'équipement, exigences de QoS) sont supposées connues parfaitement du moteur cognitif.
- Le moteur cognitif agit sur une plateforme permettant une reconfiguration quasi instantanée. Le développement d'une telle plateforme est un processus laborieux aussi avons-nous préféré utiliser un simulateur analytique pour modéliser les performances du système de communication reconfigurable (*cf.* annexe A).
- Nous avons choisi d'écarter les questions liées aux aspects protocolaires et à la nature distribuée des prises de décision. Le moteur cognitif est supposé connecté directement à tous les équipements impliqués dans la communication.

2.3 Conception d'un moteur cognitif

Le moteur cognitif exploite les possibilités de reconfiguration du terminal pour répondre au problème de conception posé par son contexte opérationnel. Il recherche une solution au problème dans son *espace de conception* délimité par les caractéristiques architecturales de l'équipement reconfigurable (*e.g.* schémas de modulation et de codage disponibles, puissance maximale de l'amplificateur, portions du spectre accessibles). Il est guidé dans son exploration par un ensemble d'objectifs opérationnels définis sur un espace appelé *espace des objectifs*. Cet espace a été particulièrement bien étudié et théorisé par Rondeau dans sa thèse [19]. Le problème de décision y est décrit comme un problème d'optimisation multi-objectifs [69] impliquant les objectifs suivants :

- Minimiser le taux d'erreurs binaires
- Maximiser le rapport signal/interférence plus bruit (*i.e.* éviter les interférences)
- Maximiser le débit d'information (throughput)
- Maximiser le débit de transmission (data rate)
- Minimiser le délai de communication et ses fluctuations (jitter)
- Minimiser la complexité algorithmique des traitements appliqués
- Minimiser la puissance consommée
- Minimiser les interférences générées
- Maximiser la bande spectrale occupée
- Maximiser l'efficacité spectrale

Ces objectifs interagissent fortement, souvent de manière conflictuelle [19, 70]. Par exemple, pour minimiser le taux d'erreurs binaires, il est préférable de transmettre les informations à une puissance d'émission élevée afin d'améliorer le rapport signal sur bruit (SNR ou Signal-to-Noise Ratio) à la réception. Cette action a cependant un impact négatif sur la consommation énergétique. De plus, elle s'oppose aux objectifs de coexistence, l'énergie rayonnée augmentant le niveau d'interférences moyen pour la région spectrale considérée. Au lieu d'augmenter la puissance d'émission, le moteur cognitif peut aussi choisir une modulation plus robuste mais cela affectera le débit de la communication dans ce cas.

Nous voyons à la lumière de ces exemples simples qu'il n'existe pas de solution unique mais une multitude de solutions en compétition. Chaque solution représente un compromis plus ou moins intéressant vis-à-vis des objectifs opérationnels.

Pour choisir la meilleure solution possible, le moteur cognitif a besoin d'organiser ses actions en deux phases essentielles. Nous présentons ces deux phases dans un premier temps puis nous résumons la mission du moteur cognitif à l'aide d'un schéma récapitulatif.

2.3.1 Détermination du problème de conception

Le moteur cognitif identifie le problème de conception en analysant les informations qu'il récupère sur son contexte. Il exploite ces informations pour réduire l'espace de conception, identifier les objectifs opérationnels et modéliser l'environnement radio.

Réduction de l'espace de conception

L'espace de conception peut être réduit pour deux raisons. Premièrement, le cadre de régulation impose des contraintes sur la puissance maximale d'émission et sur les portions du spectre accessibles. Le moteur cognitif tient compte des politiques de régulation en vigueur en rejetant les formes d'onde trop gourmandes en ressources spectrales. Deuxièmement, la radio peut avoir à communiquer avec un équipement moins flexible. Le moteur cognitif s'adapte à cette contrainte matérielle en éliminant les formes d'onde indisponibles de son espace de conception.

Spécification des objectifs opérationnels

La spécification des objectifs opérationnels est une tâche importante puisqu'elle définit la notion d'optimalité guidant le processus d'exploration appliqué lors de la résolution du problème. Le moteur cognitif est influencé par plusieurs facteurs lors de cette phase :

- Chaque application est caractérisée par des besoins différents en termes de QoS. Les communications vocales, par exemple, sont sensibles au délai de communication alors qu'un transfert de fichier nécessite un débit important. Le moteur cognitif consulte le profil de l'application sélectionnée afin de récupérer les consignes d'optimisation³ ainsi que les contraintes quantitatives imposées sur les indicateurs de performance clés⁴.
- Le moteur cognitif tient compte de l'environnement radio lors de la spécification des contraintes. Le profil de l'application peut définir plusieurs niveaux de QoS plus ou moins exigeants sur la qualité du lien radio. Une vidéo peut ainsi être encodée en haute définition, en qualité réduite, voire en noir et blanc en fonction des conditions de propagation.
- Le moteur cognitif est sensible aux préférences de l'utilisateur et de l'opérateur. Les exigences de QoS sont à la mesure du prix payé. On peut donc imaginer que les utilisateurs les plus riches seront favorisés lors du partage de ressources. Les équipes d'intervention d'urgence seront aussi prioritaires étant donnée l'importance de leurs communications. L'opérateur a néanmoins un souci d'équité, notamment entre les utilisateurs de même "classe". Il va donc influencer la phase de conception des radios en fonction de la capacité actuelle du réseau. Si le réseau est chargé, il indique au moteur cognitif de privilégier des configurations économes en ressources radio (faible puissance d'émission, modulation bande étroite). Le moteur cognitif peut chercher à améliorer la QoS dans le cas contraire.
- Les objectifs sont modulés en fonction du niveau d'énergie restant dans les batteries des équipements portables. Les configurations simples sont de plus en plus avantageuses au fur et à mesure que le terminal se décharge. Le moteur cognitif peut alors compenser la dégradation de QoS en augmentant la puissance d'émission de la station de base (non contrainte en énergie) dans les limites imposées par le cadre de régulation.

3. Si l'application est gourmande en bande passante, le moteur cognitif favorise les formes d'onde s'étalant sur une large bande fréquentielle avec une modulation d'ordre élevé et un faible taux de redondance.

4. Il est souvent nécessaire de garantir une fiabilité minimum du lien radio en maintenant un taux d'erreurs binaires en-dessous d'une valeur cible (*e.g.* 10^{-6} pour un transfert de données).

Modélisation de l'environnement radio

L'environnement radio a un impact majeur sur les performances des configurations et les exigences de QoS sont plus ou moins difficiles à atteindre en fonction des conditions de propagation. L'égalisation est, par exemple, nécessaire lorsque le canal fait apparaître de l'interférence inter-symboles. Si le canal n'est pas sélectif, elle devient plus handicapante qu'utile en raison de la complexité des traitements appliqués.

Le moteur cognitif a besoin de modéliser l'environnement radio afin d'estimer l'adéquation des configurations aux objectifs de QoS. Le modèle est voulu aussi fidèle et compact que possible afin de guider le raisonnement logique avec efficacité tout en limitant la complexité du processus de décision. Le canal radio est un milieu de propagation que l'on peut caractériser par de nombreuses mesures (*e.g.* rapport signal sur interférences plus bruit, réponse impulsionnelle du canal). Ces mesures sont traditionnellement utilisées par le récepteur (*e.g.* égalisation) et l'émetteur (*e.g.* adaptation de modulation et de codage). L'analyse du spectre radio requiert en revanche plus d'efforts de recherche car les radios actuelles ne sont dotées d'aucune sensibilité spectrale, l'accès au spectre étant contrôlé par un processus centralisé et déterministe (*i.e.* la station de base dispose d'un nombre limité de canaux physiques alloués dynamiquement en fonction de leur disponibilité).

2.3.2 Résolution du problème de conception

Une fois le problème de conception identifié, il est nécessaire de le résoudre. Le moteur cognitif va alors explorer l'espace de conception à la recherche de la configuration la plus en adéquation avec ses objectifs opérationnels [71, 72]. Cette exploration est réalisée en ligne donc elle doit être rapide et efficace afin de pouvoir suivre les variations du contexte. Le moteur cognitif répond en effet à un problème d'optimisation temps réel. Son objectif n'est pas de trouver la solution optimale à tout prix mais de produire la meilleure solution possible dans un temps acceptable [19]. Pour se rapprocher plus rapidement du comportement optimal, le moteur cognitif a donc tout intérêt à exploiter au maximum l'ensemble des connaissances disponibles (connaissances expertes, résultat des phases d'exploration antérieures). Lors du processus d'exploration, le moteur cognitif peut être amené à tester ses meilleures propositions à la volée afin de les évaluer et de les comparer. Toute mauvaise décision a un impact négatif sur l'expérience de l'utilisateur. Le moteur cognitif doit donc savoir réagir rapidement après une erreur tout en apprenant à minimiser ces situations d'échecs avec le temps.

La phase d'analyse est réalisée en permanence pour actualiser le problème de conception au contexte. En revanche, la seconde phase n'est déclenchée qu'en cas de nécessité car il est préférable de réduire son coût au minimum. Les mécanismes d'exploration impliqués dans la sélection d'une nouvelle configuration (raisonnement et apprentissage) introduisent une complexité additionnelle drainant l'énergie de la batterie. La configuration du lien radio passe par des efforts de signalisation (handshaking⁵) diminuant le débit d'informations utiles. Enfin, la plateforme matérielle peut avoir des problèmes à suivre des ordres de reconfiguration trop fréquents sans introduire une latence importante affectant l'expérience de l'utilisateur. Un processus de surveillance est donc nécessaire à ce titre.

5. Les objets communicants ont besoin de se mettre d'accord sur les paramètres communs d'émission/réception (*e.g.* bande fréquentielle sélectionnée, modulation employée, code correcteur d'erreurs appliqué).

2.3.3 Synoptique d'un moteur cognitif

Les réflexions présentées dans cette section ont guidé la conception de notre moteur cognitif. La représentation générale du problème est résumée à la figure 2.3. Le moteur cognitif part d'informations récupérées sur son contexte opérationnel pour déterminer le problème de conception (analyse du contexte), le résoudre (exploration de l'espace de conception) et ainsi trouver une configuration adaptée. Le processus de surveillance décide quand il est nécessaire de présenter le problème de conception (espace de conception / environnement radio / espace des objectifs) au processus d'exploration.

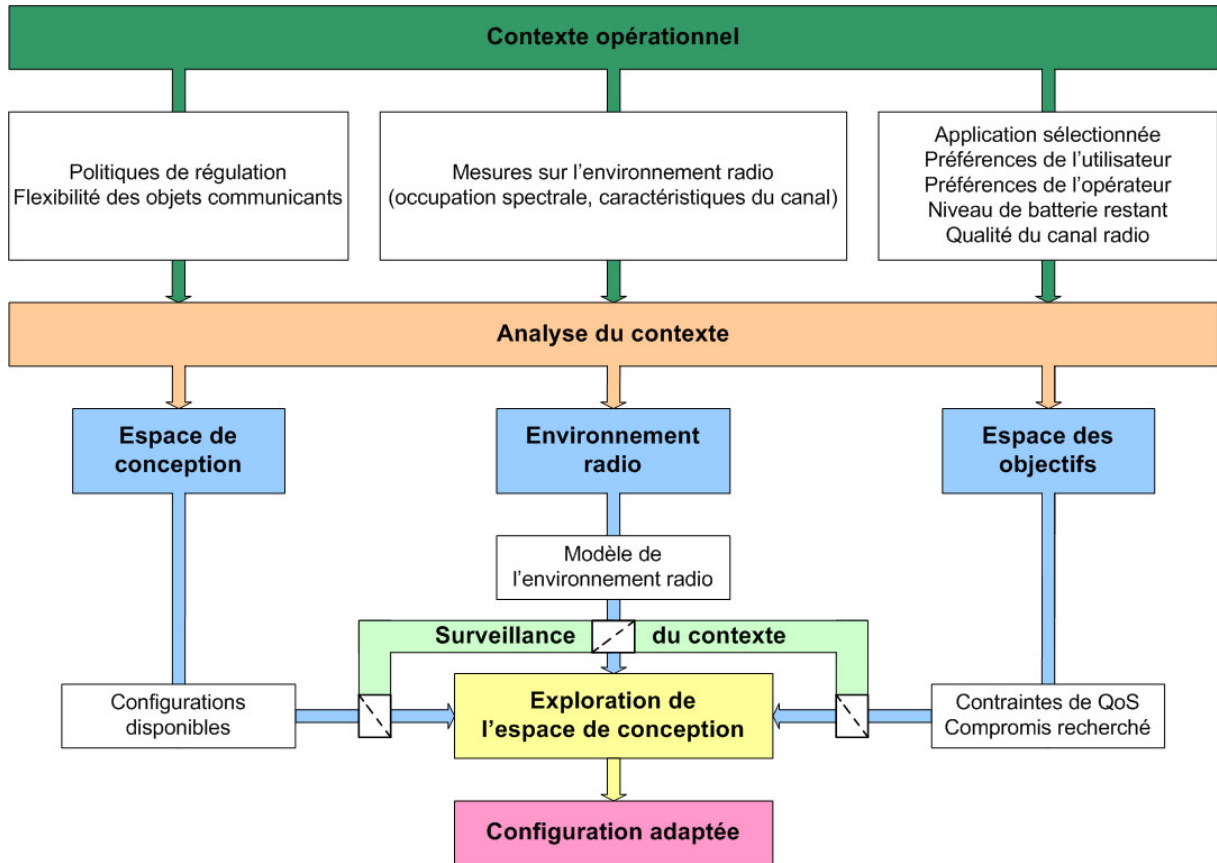


FIGURE 2.3 – Synoptique d'un moteur cognitif

2.4 Approches cognitives existantes

L'état de l'art commence à s'étoffer de nombreuses solutions que nous proposons de classer en trois approches.

L'*approche experte* se focalise sur la définition de règles guidant la conception dynamique de la radio. Le moteur cognitif fait appel au moteur d'inférence pour raisonner sur son contexte et proposer une stratégie de communication dans le respect du cadre de régulation.

L'*approche exploratoire* repose sur l'utilisation d'algorithmes génétiques [73] pour faire évoluer le comportement de la radio en fonction du problème d'optimisation multi-objectifs posé par le contexte opérationnel. Les algorithmes génétiques s'appuient sur les principes de l'évolution et

de la sélection naturelle pour guider la recherche d'une solution adaptée dans l'espace complexe défini par le problème de conception [34, 19].

L'*approche prédictive* propose d'intégrer des capacités de prédiction dans le moteur cognitif afin de lui permettre d'estimer en temps-réel l'impact de l'environnement radio sur les performances de la communication.

Pour chaque approche, nous revisitons le cycle cognitif présenté à la section 2.1 puis nous analysons les différentes solutions suivant quatre critères :

- Comment sont représentées les connaissances ?
- Comment sont-elles exploitées ?
- Comment est réalisé l'apprentissage (incrémental, non incrémental ou assisté) ?
- Quelles sont les connaissances apprises ?

2.4.1 Approche experte

L'approche experte profite des connaissances établies en télécommunications pour définir des modèles, des règles et des politiques permettant une prise de décision autonome par la radio. Les équipements se comportent de manière déterministe sans qu'aucune créativité ne leur soit accordée (raisonnement de premier niveau uniquement). Leurs performances dépendent directement de la qualité des connaissances embarquées et de leur pertinence vis-à-vis des situations rencontrées. La radio s'appuie sur un système d'inférence pour prendre des décisions logiques en adéquation avec les observations réalisées. En fonction de la complexité du problème, le système d'inférence peut appliquer de nombreuses règles en série et/ou en parallèle avant de pouvoir aboutir à une configuration légale et adaptée. Pour permettre un raisonnement efficace, les règles/politiques sont exprimées dans un langage structuré, reposant sur une ontologie du domaine radio. Une ontologie est un mécanisme formel permettant de capturer la terminologie et les connaissances d'un domaine d'intérêt ainsi que les relations entre les différents concepts impliqués [74]. Une radio opportuniste doit, par exemple, pouvoir raisonner sur des concepts aussi variés que les bandes fréquentielles, les canaux radio, les conditions de propagation, les interférences et le bruit, les formes d'onde, les frontières géopolitiques, les lieux géographiques, le temps et les types de service [75].

Mitola développe le premier langage dédié (Radio Knowledge Representation Language) lors de sa thèse [35]. L'approche experte a connu un grand essor depuis [76, 77, 7], notamment sous l'impulsion du projet XG (neXt Generation) soutenu par le DARPA (Defense Advanced Research Projects Agency) [78, 79, 80, 81].

Pour traduire l'ontologie dans un format compréhensible par la machine, les experts humains s'appuient sur des langages du web sémantique dérivés du XML (eXtensible Markup Language) tels que le RDF (Resource Description Framework) ou l'OWL (Web Ontology Language). La construction d'une ontologie efficace n'est pas un exercice facile. L'ontologie doit capturer la richesse de la sémantique radio tout en permettant un raisonnement rapide. Elle doit aussi pouvoir évoluer facilement afin d'intégrer de nouvelles informations apparaissant avec l'évolution des technologies radios. Le raisonnement sera rendu de plus en plus complexe avec la multiplication d'informations spécifiques et il se peut que plusieurs règles rentrent en conflit pour une situation donnée. Pour résoudre les ambiguïtés dans la base de connaissance, les concepteurs du système peuvent associer une priorité à chaque règle. Ils peuvent aussi développer des métapolitiques (politiques sur des politiques) et les utiliser pour détecter les conflits sémantiques entre les règles [75].

L'approche experte s'inscrit dans la lignée des méthodes traditionnelles d'adaptation, d'où sa popularité. L'adaptation de modulation et de codage se base en effet sur des règles simples

(logique de premier ordre) définies manuellement par les concepteurs du système à partir de campagnes de mesures, de modèles analytiques et/ou de simulations intensives. Cette méthodologie a été poursuivie avec des radios plus flexibles [82]. L'arrivée de la radio cognitive change la donne sur le plan technique car la logique de premier ordre doit laisser place à des mécanismes dynamiques, plus évolués (logiques d'ordre supérieur) afin de permettre un raisonnement plus complet dans un espace plus complexe. Nous noterons cependant que la méthodologie reste la même sur le plan conceptuel.

L'approche experte est nécessaire pour respecter le cadre de régulation. Mais elle ne présente pas un intérêt aussi flagrant pour la mise au point d'une stratégie de communication. Elle nécessite, en effet, un effort important de la part des concepteurs humains pour la standardisation du langage et la mise au point de règles de conception. Or l'effort est proportionnel à la complexité du système conçu et nous avons vu dans la section 2.3 que l'espace de décision d'une radio cognitive était fort complexe (prise en compte de plusieurs objectifs conflictuels, multiplication des configurations disponibles, diversité des contextes opérationnels). Le comportement des radios peut certes être amélioré par une boucle hors-ligne capable d'analyser et d'extraire des informations par simulation pour mettre au point de nouvelles règles. Mais cet apprentissage assisté présente des inconvénients :

- La radio ne sait pas réagir comme il faut lorsqu'elle opère dans des environnements électromagnétiques non favorables et non familiers [34].
- La mise à jour des règles est moins fréquente.
- Les règles de conception sont optimisées pour le plus grand nombre plutôt que pour une radio particulière.
- La base de connaissance peut se retrouver alourdie par des règles de faible pertinence relativement aux habitudes de l'utilisateur. Ces règles compliquent inutilement le processus de raisonnement.
- La dissémination des règles consomme des ressources spectrales.

Il serait donc préférable que la radio apprenne à développer en ligne sa propre base de connaissances en fonction des expériences vécues [83].

La figure 2.4 représente le cycle cognitif typique d'une approche experte. Les réponses aux quatre questions posées en introduction sont rassemblées dans le tableau 2.1.

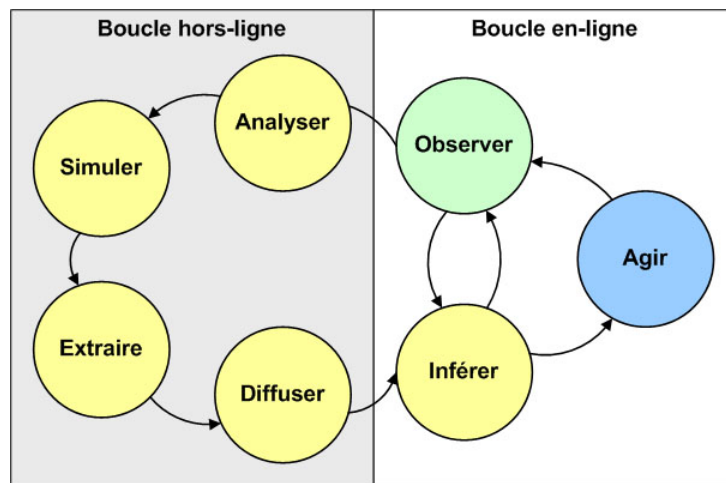


FIGURE 2.4 – Cycle cognitif pour l'approche experte

<i>Représentation des connaissances :</i>	Règles/politiques
<i>Méthode d'exploitation des connaissances :</i>	Moteur d'inférence
<i>Type d'apprentissage :</i>	Assisté
<i>Connaissances apprises :</i>	Règles de conception

TABLE 2.1 – Spécificités de l'approche experte

2.4.2 Approche exploratoire

L'approche exploratoire est née avec l'université de Virginia Tech qui a reconnu et démontré l'intérêt des algorithmes génétiques pour l'optimisation en temps réel des performances de la radio [34, 6, 84, 85, 86, 30, 22, 70, 87, 19, 88, 89, 90, 91]. Les travaux du département CWT (Cognitive Wireless Technology) de Charles Bostian ont inspirés également de nombreux organismes de recherche [92, 93, 68, 94]

Les algorithmes génétiques [73] s'appuient sur un modèle d'optimisation inspiré de la théorie darwinienne de l'évolution pour guider progressivement une population de solutions potentielles vers les régions les plus intéressantes de l'espace de recherche. Une configuration particulière de la radio est encodée dans un chromosome où chaque gène représente une fonction de traitement du signal (*e.g.* codage, modulation) ou un paramètre spécifique (*e.g.* vitesse de modulation, puissance d'émission). Pour répondre au problème de conception posé, les algorithmes génétiques manipulent une population de configurations admissibles (*i.e.* en accord avec le cadre de régulation) sur plusieurs générations en appliquant différents opérateurs génétiques jusqu'à vérifier un critère d'arrêt prédéfini. Les performances des individus sont évaluées à chaque génération à l'aide de fonctions d'évaluation (fitness function) caractéristiques du problème. L'opérateur de sélection favorise les individus les plus adaptés en augmentant leur chance de survie d'une génération à une autre. La population est ensuite diversifiée à l'aide d'opérateurs de variations appliqués aléatoirement. Le croisement de deux chromosomes parents produit deux chromosomes fils incorporant chacun du matériel génétique pris dans le patrimoine initial. Certains individus peuvent aussi subir des mutations. Le cycle évaluation - sélection - croisement - mutation est répété itérativement jusqu'à obtenir une configuration satisfaisante vis-à-vis des objectifs identifiés. La procédure peut aussi être arrêtée prématurément pour tenir compte des contraintes de prise de décision temps réel.

Les algorithmes génétiques présentent un intérêt évident pour la radio cognitive car ils permettent d'explorer efficacement un large espace de recherche. Cet avantage s'accompagne néanmoins de trois inconvénients.

Complexité des mécanismes

Malgré leur efficacité, les algorithmes génétiques introduisent une certaine complexité dans le terminal radio ce qui nuit à la réactivité du moteur cognitif (convergence lente des algorithmes) et à l'autonomie énergétique des équipements. La phase d'évaluation requiert en effet de nombreux calculs qui sont répétés pour toutes les solutions à chaque génération. Deux méthodes ont été proposées pour réduire cet impact. La première méthode s'appuie sur les propriétés de traitement parallèle des algorithmes génétiques pour distribuer la charge de calcul entre plusieurs radios [68]. La seconde méthode s'inspire de la théorie du raisonnement par cas [95, 96]. Le moteur cognitif entretient une base de cas passés afin de pouvoir s'y référer lorsqu'il rencontre des cas similaires. Chaque cas mémorisé contient la description du problème de conception (objectifs

imposés, environnement radio, contraintes de conception) ainsi qu'un ensemble de configurations prometteuses obtenues lors d'une précédente recherche. Lorsqu'un nouveau problème est posé, le moteur cognitif identifie le cas le plus pertinent en considérant sa précédente utilité et sa similarité au cas actuel. Les configurations associées au cas vont alors former la population initiale utilisée par les algorithmes génétiques. Cette population initiale est souvent bien plus proche de la solution optimale qu'une population aléatoire. Les algorithmes génétiques convergent donc plus rapidement vers une solution acceptable. Grâce à cette théorie, le moteur cognitif dispose de capacités d'apprentissage donc il ne compte plus seulement sur la puissance d'optimisation des algorithmes génétiques. En pratique, le raisonnement par cas permet un gain moyen autour de 5% quant aux performances des algorithmes génétiques [19].

Sensibilité aux paramètres

Le second inconvénient est lié à la sensibilité de la méthode quant au choix des paramètres. L'évolution va en effet connaître plus ou moins de succès en fonction des valeurs choisies pour la taille de la population, le taux de croisement ou le taux de mutation. Le critère d'arrêt joue aussi sur le compromis complexité/performances des algorithmes génétiques. Une adaptation en ligne de ces paramètres pourraient contribuer à améliorer l'efficacité du moteur cognitif [19].

Dépendance à un modèle analytique

Lorsqu'il est confronté à un nouveau problème, le moteur cognitif simule l'évolution du comportement de la radio avant de tester la configuration la plus prometteuse. Cette expérimentation virtuelle est réalisée en parallèle de la communication donc la radio n'est pas perturbée par les essais répétés du moteur cognitif. C'est un avantage en première analyse. Nous notons cependant que la qualité de l'expérimentation dépend fortement de la fidélité du modèle de simulation vis-à-vis de la situation rencontrée. Lors de la phase d'évaluation, les performances des configurations sont estimées à l'aide d'expressions issues de modèles analytiques. Comme l'a souligné Baldo [31], ces modèles présentent des inconvénients dans un contexte de radio cognitive :

- Ils sont basés sur des hypothèses de modélisation (*e.g.* idéalisation du canal) qui ne s'appliquent pas toujours pour les scénarios de la vie réelle.
- Les résultats du modèle peuvent être biaisés par rapport aux performances réelles à cause du comportement non idéal de l'équipement, de la défaillance de certains composants ou encore de la présence de facteurs environnementaux non prévus.
- Des modèles analytiques ne sont pas disponibles pour toutes les situations.
- Certains modèles ne peuvent pas être implémentés dans le moteur cognitif sans introduire une complexité excessive limitant l'intérêt des algorithmes génétiques.
- A chaque fois que la radio intègre un nouveau composant, il est nécessaire de développer hors-ligne un nouveau modèle analytique puis de le charger dans le système. C'est un inconvénient majeur avec la flexibilité grandissante des équipements et la diversité des environnements radio amenés à être rencontrés. Cet inconvénient est d'autant plus fort si le développement du modèle requiert un effort humain important.

Pour s'affranchir de ces problèmes, la radio cognitive doit intégrer des capacités de prédiction afin d'estimer l'impact réel de l'environnement radio sur les performances des configurations. L'intérêt d'un modèle prédictif apparaît plus clairement à la lecture de [33].

Comme pour l'approche experte, nous revisitons le cycle cognitif générique à la figure 2.5 et nous résumons la réponse aux quatre questions de l'introduction dans le tableau 2.2.

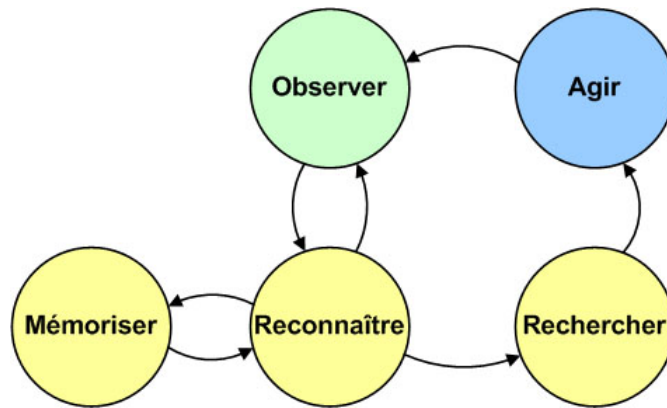


FIGURE 2.5 – Cycle cognitif pour l'approche exploratoire

<i>Représentation des connaissances :</i>	Fonctions d'évaluation Population initiale de configurations potentielles
<i>Méthode d'exploitation des connaissances :</i>	Algorithmes génétiques
<i>Type d'apprentissage :</i>	Incrémental
<i>Connaissances apprises :</i>	Situations typiques mémorisées

TABLE 2.2 – Spécificités de l'approche exploratoire

2.4.3 Approche prédictive

L'approche prédictive cherche à établir une relation fonctionnelle entre les mesures du système, ses paramètres et ses performances. De nombreuses techniques ont été proposées à cette fin. Nous pouvons citer, par exemple, les modèles de Markov cachés [84,97], les modèles de régression [29,98,99,100], les arbres de décision [101,99], les réseaux bayésiens [102,103], les machines à vecteurs de support [104] ou les réseaux de neurones [31,101].

Les capacités de prédiction améliorent la fiabilité du moteur cognitif en lui permettant de raisonner à partir d'estimations robustes sur les performances des configurations disponibles. La qualité des décisions prises dépend alors de deux facteurs :

- *Le pouvoir représentatif de la technique utilisée :* La technique sélectionnée doit pouvoir représenter la relation fonctionnelle cible ou du moins l'approcher de près. Tout biais de représentation se traduit par une dégradation de la précision des prédictions ce qui peut induire le moteur cognitif en erreur. Par exemple, un modèle de régression linéaire est mis en défaut pour des relations non linéaires entre les entrées/sorties. Des modèles de régression plus complexes peuvent aussi se retrouver dans cette situation. Dans [98,99,100], Weingart *et al.* proposent une méthode de reconfiguration dynamique basée sur l'analyse statistique des relations de cause à effet entre les paramètres de reconfiguration (facteurs) et les performances du système (réponses). Les auteurs s'appuient sur la méthodologie DOE (Design of Experiments) pour identifier les facteurs (ou combinaison de facteurs) affectant les réponses observées de manière significative. Un modèle prédictif est dérivé dans un second temps afin de permettre au moteur cognitif de sélectionner une configuration en accord avec les spécifications de QoS. Ce modèle n'est pas toujours fiable [99,100]. Les auteurs ont ainsi recours à un arbre de régression pour quantifier l'impact de certains paramètres sur la latence de communication.

- *La fiabilité du modèle prédictif vis-à-vis de l'expérience rencontrée* : Le modèle prédictif est construit par apprentissage à partir d'un nombre limité d'échantillons. Pour produire un modèle fiable, l'algorithme d'apprentissage doit disposer de données suffisamment nombreuses et représentatives. Dans le cas contraire, le moteur cognitif peut être amené à prendre des décisions sur des situations pour lesquelles il n'a que peu ou pas d'expérience.

L'apprentissage est réalisé hors ligne dans un premier temps pour que le moteur cognitif débute avec une expérience acceptable. La fiabilité du modèle dépend alors du plan d'expérience mis en place pour construire la base d'apprentissage mais aussi des propriétés de généralisation de la technique utilisée. L'analyse statistique de Weingart *et al.* requiert, par exemple, une série intensive d'expérimentations pour tester toutes les combinaisons possibles des facteurs [100]. Mais le modèle produit est-il toujours aussi fiable pour une situation différente de celles testées ? Ce problème lié au pouvoir représentatif du modèle est connu sous le nom de dilemme biais-variance : si le modèle possède un nombre de paramètres excessif, sa sortie peut passer avec une très grande précision par tous les points d'apprentissage, mais fournir des résultats dépourvus de signification entre ces points ; s'il possède un nombre de paramètres trop restreint, le modèle n'est pas suffisamment riche pour rendre compte de la complexité de la fonction de régression inconnue [105].

La base d'apprentissage initiale ne saura jamais couvrir tous les contextes opérationnels rencontrés par une radio donc le moteur cognitif doit pouvoir réaliser un apprentissage en ligne afin d'ajuster son modèle en fonction des modifications des objectifs et des variations de l'environnement radio. Cet apprentissage n'est possible qu'à deux conditions :

- La phase d'apprentissage ne doit pas être handicapante pour l'autonomie énergétique des équipements. Weingart *et al.* reconnaissent eux-mêmes que leur méthode est trop exigeante en capacités de calcul pour être implémentée sur un système temps-réel. Leur solution n'est donc pas encore viable pour une radio cognitive. Mais la méthodologie développée reste intéressante car elle permet d'extraire des connaissances qualitatives sur les relations fonctionnelles entre les facteurs et les réponses.
- Le moteur cognitif doit pouvoir conduire une série d'expérimentations en ligne afin de récolter des informations pertinentes pour la mise à jour du modèle. En effet, la radio ne peut pas se contenter d'exploiter ses connaissances pour se rapprocher du comportement optimal. Elle a besoin de faire des essais et d'apprendre de ses expériences. Le moteur cognitif doit par contre proposer un plan d'expérience intelligent afin de minimiser l'impact de ses tentatives lors de la communication.

Nous verrons que notre moteur cognitif répond à ces deux conditions.⁶

Dans [31], Baldo *et al.* utilisent des réseaux de neurones pour caractériser les performances de communication d'une radio dans son contexte. Les réseaux de neurones [105] présentent un intérêt indéniable pour cette tâche. Cet outil de modélisation par apprentissage est capable d'ajuster des fonctions non linéaires très générales à partir d'observations sur les entrées (*e.g.* rapport signal sur bruit, dispersion fréquentielle du canal) et les sorties (*e.g.* taux d'erreurs binaires, débit, latence) du système. La construction du modèle requiert très peu de connaissances *a priori* sur les lois du système (modélisation boîte noire) donc sa validité ne dépend pas d'une idéalisation excessive du problème (*e.g.* hypothèses simplificatrices, idéalité des composants matériels et logiciels). De plus, le traitement parallèle des informations permet au réseau de faire ses prédictions rapidement [106].

6. Les modèles prédictifs utilisés peuvent être mis à jour de manière incrémentale en fonction des expériences réalisées et des résultats observés. L'espace de conception est exploré par essais/erreurs sous la direction d'un algorithme de supervision cherchant à optimiser les connaissances du moteur cognitif sur son environnement tout en réduisant les retombées négatives de l'expérimentation sur la communication.

Le moteur cognitif peut donc suivre les variations du canal en temps réel.

Le cycle cognitif typique d'une approche prédictive est présenté à la figure 2.6. Étant donné la diversité des solutions, nous ne donnons que des réponses générales aux questions de l'introduction (tableau 2.3).

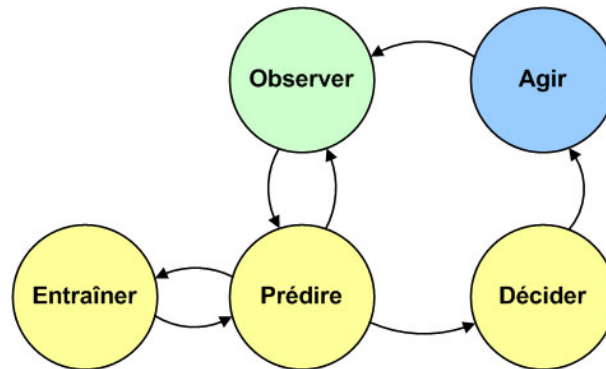


FIGURE 2.6 – Cycle cognitif pour l'approche prédictive

<i>Représentation des connaissances :</i>	Paramètres du modèle prédictif
<i>Méthode d'exploitation des connaissances :</i>	Prédiction des performances et sélection d'une configuration en adéquation avec les objectifs
<i>Type Apprentissage :</i>	Incrémental/Non incrémental/Assisté
<i>Connaissances apprises :</i>	Relation fonctionnelle entre les mesures du système, ses paramètres et ses performances

TABLE 2.3 – Spécificités de l'approche prédictive

2.5 Approche adoptée dans nos travaux

Notre approche est prédictive par nature mais nous reprenons aussi des éléments conceptuels des deux autres approches. De l'approche experte, nous reconnaissons le besoin d'embarquer des connaissances dans le moteur cognitif afin de l'aider dans ses prises de décision. De l'approche exploratoire, nous retenons l'importance d'explorer l'espace de conception pour optimiser le comportement de la radio.

Nous avons conçu notre moteur cognitif dans le but de rendre la radio pleinement autonome face à son contexte opérationnel. Nous avons notamment développé les quatre points suivants :

- Le moteur cognitif ne peut pas compter sur une idéalisation mathématique du problème pour prendre ses décisions. Il se base sur un modèle prédictif pour pleinement appréhender son contexte opérationnel sans qu'aucune hypothèse ne soit faite sur les lois régissant le système radio.
- La radio doit pouvoir améliorer ses performances au fur et à mesure des expériences rencontrées afin de se rapprocher progressivement d'un comportement optimal. L'espace de conception est exploré en ligne de manière à récolter des informations fiables pour la mise à jour du modèle. Le moteur cognitif fait des essais et il apprend de ses erreurs pour éviter de les reproduire. Il construit son plan d'expérience dynamiquement en fonction de la

confiance qu'il porte dans ses connaissances mais aussi en fonction des retours du système. Il sait aussi arrêter son expérimentation quand elle devient nuisible pour la QoS délivrée à l'utilisateur. Grâce à cette méthodologie, la radio devient de plus en plus efficace face aux situations fréquemment rencontrées tout en restant capable de s'adapter à de nouvelles situations.

- Le modèle prédictif a besoin d'évoluer en temps réel pour suivre les contraintes immédiates de la radio. Le moteur cognitif analyse en permanence les informations récoltées par le terminal pour les transformer en connaissances intégrables dans le modèle. Cet apprentissage incrémental a l'avantage de garantir une fiabilité constante du modèle prédictif. Le moteur cognitif peut donc s'adapter à de nouvelles situations avec une bonne réactivité. La contrainte temps-réel impose cependant de développer des mécanismes d'apprentissage légers pour préserver l'autonomie de la batterie et ne pas perturber la communication (partage des ressources de calcul) [83].
- Le moteur cognitif peut bénéficier de connaissances expertes pour améliorer la fiabilité des mécanismes, augmenter leur rapidité et/ou réduire leur complexité. Notre algorithme de supervision exploite des connaissances qualitatives sur les configurations pour réaliser une exploration astucieuse de l'espace de conception et ainsi minimiser les erreurs d'expérimentation.

Notre cycle cognitif est présenté à la figure 2.7. Nous y détaillons les différentes actions associées à notre modèle cognitif ainsi que leurs interactions. La figure met aussi en valeur les quatre processus cognitifs⁷ haut-niveau embarqués dans la radio. Les détails techniques des quatre processus sont présentés dans la suite du document. Nous nous limitons ici à une description conceptuelle de notre approche afin de faciliter la compréhension des mécanismes proposés.

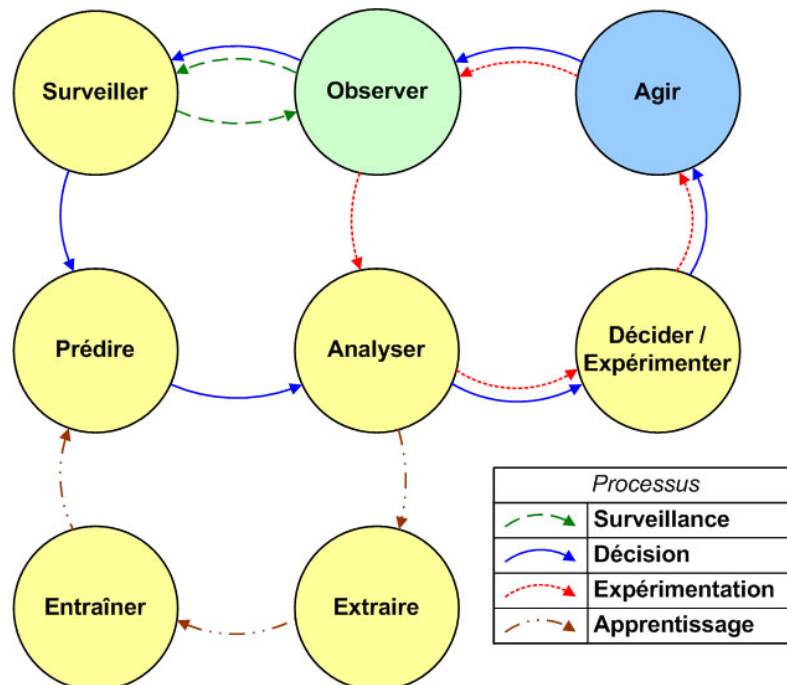


FIGURE 2.7 – Cycle cognitif adopté

7. Un processus cognitif est un enchaînement d'opérations en relation avec la saisie des informations, leur stockage et/ou leur traitement.

Le *processus de surveillance* est activé continuellement. Il se charge d'analyser les fluctuations de l'espace de décision afin d'identifier une opportunité de maintenir ou d'améliorer les objectifs opérationnels. Son objectif est de minimiser la complexité du moteur cognitif tout en garantissant une reconfiguration efficace de la radio.

Le *processus de décision* est déclenché lorsqu'une reconfiguration s'avère nécessaire. Le moteur cognitif se retrouve alors face au problème de conception dérivé par le processus de surveillance après analyse du contexte opérationnel. Il réduit l'espace de conception une première fois pour prendre en compte les contraintes matérielles et les directives de régulation (voire de l'opérateur). Il se base ensuite sur son modèle prédictif pour identifier quelles sont les configurations susceptibles de vérifier les contraintes de QoS étant données les conditions de propagation observées. L'espace de conception est ainsi réduit une seconde fois, le moteur cognitif écartant les configurations trop peu performantes pour délivrer le service demandé. Le moteur cognitif peut ensuite sélectionner la configuration la plus satisfaisante parmi les configurations restantes. Il peut aussi décider de vérifier l'intégrité de son modèle prédictif en déclenchant le processus d'expérimentation.

Le *processus d'expérimentation* permet au moteur cognitif de mettre ses connaissances à l'épreuve dans l'optique de se rapprocher du comportement optimal. Le modèle prédictif agit comme un filtre dynamique pour le processus de décision. Les configurations disponibles sont classées en fonction de leur capacité/incapacité à fournir le service demandé.

La figure 2.8 est une illustration conceptuelle de l'espace de conception tel qu'il est partitionné par notre moteur cognitif. Elle permet d'illustrer l'impact de la qualité du modèle prédictif sur la prise de décision.

Le cas (a) représente un partitionnement parfait de l'espace de conception. Le modèle prédictif réalise une classification sans erreur donc le moteur cognitif dispose de toutes les informations pour choisir la configuration la plus adaptée. Ce cas idéal suppose que le moteur cognitif a une maîtrise totale sur le problème de conception posé. Le modèle prédictif n'est cependant pas toujours aussi fiable comme en témoigne les cas (b), (c) et (d).

Le cas (b) est obtenu lorsque le modèle prédictif conduit à une réduction insuffisante de l'espace de conception. Le moteur cognitif peut alors dégrader temporairement la QoS s'il sélectionne une configuration ne vérifiant pas les contraintes de performances (mauvaise décision). Il doit dans ce cas réagir rapidement afin de pouvoir rétablir la QoS. La recherche de la configuration la plus satisfaisante peut aussi aboutir sur une configuration en phase avec les contraintes imposées. Les erreurs de classification n'ont alors aucun impact sur la qualité des décisions prises.

Le cas (c) est obtenu lorsque le modèle prédictif conduit à une réduction trop importante de l'espace de conception. Avec ce type d'erreur, le moteur cognitif est toujours garanti de prendre une décision acceptable (*i.e.* la configuration sélectionnée supporte l'application utilisée) mais il peut aussi être amené à prendre une décision sous optimale (*i.e.* une configuration écartée à tort aurait pu vérifier un meilleur compromis entre les objectifs opérationnels). Les erreurs de classification peuvent là encore n'avoir aucun impact si la configuration optimale se trouve dans l'espace de conception réduit.

Le cas (d) cumule les deux types d'erreurs. Le moteur cognitif a de plus grandes chances de prendre des mauvaises décisions si le modèle prédictif conduit à de nombreuses erreurs du premier type. Avec un nombre élevé d'erreurs du second type, le moteur cognitif risque fort de proposer une configuration beaucoup moins efficace que la configuration optimale.

La phase d'expérimentation permet au moteur cognitif de perfectionner son modèle prédictif au fur et à mesure des expériences rencontrées. Les erreurs du premier type sont repérées passivement lorsque le moteur cognitif sélectionne une configuration trop peu performante alors

★	Configuration rendue indisponible (régulation, opérateur, équipement moins flexible)
◆	Configuration compatible avec les exigences de QoS
●	Configuration incompatible avec les exigences de QoS
	Ensemble des configurations éliminées par la première opération de réduction
	Ensemble des configurations prédites comme compatibles par le modèle
	Ensemble des configurations prédites comme incompatibles par le modèle

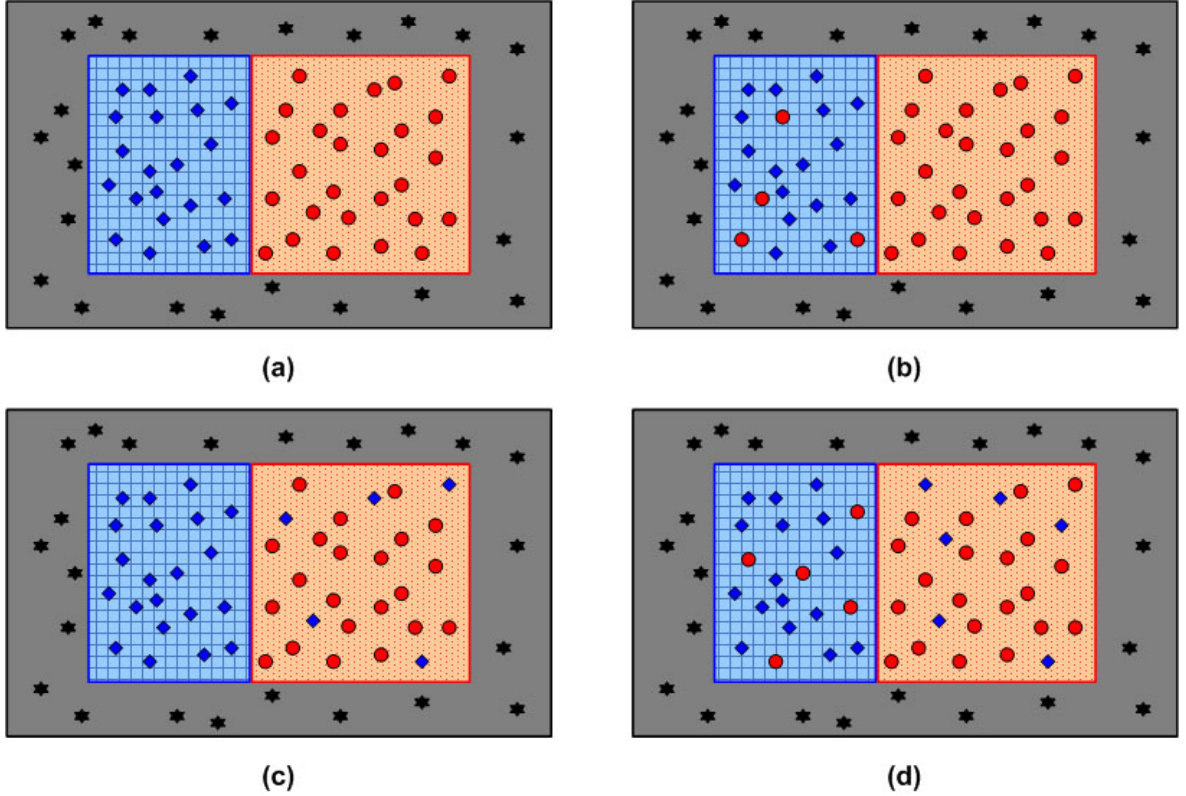


FIGURE 2.8 – Partitionnement de l'espace de conception pour différentes qualités du modèle prédictif (a, b, c, d)

qu'il estime proposer une solution compatible avec le problème (cercle rouge dans la zone quadrillée bleue). Les erreurs du second type nécessitent en revanche une expérimentation active. Le moteur cognitif peut ainsi décider de tester des configurations prédites comme incompatibles pour mettre ses connaissances à l'épreuve et tenter de se rapprocher du comportement optimal (recherche des losanges bleus dans la zone pointillée rouge). L'expérimentation est réalisée en ligne donc il est nécessaire de maîtriser l'exploration offensive du moteur cognitif pour ne pas trop dégrader la QoS. L'algorithme de supervision sera efficace s'il permet de repérer rapidement les losanges bleus en évitant au maximum de tester les cercles rouges.

Le *processus d'apprentissage* est déclenché après chaque expérimentation. Le moteur cognitif analyse les résultats de l'expérimentation afin d'extraire des informations pour la mise à jour du modèle prédictif. Il consolide ses connaissances si l'expérience de conception réalisée ne permet pas de mettre le modèle en défaut. Il les corrige dans le cas contraire. De cette manière, la fiabilité des systèmes prédictifs augmente progressivement et le moteur cognitif prend de l'assurance sur les problèmes qu'il a l'habitude de traiter.

L'outil de visualisation exposé à la figure 2.8 nous a permis de mettre en évidence le besoin d'expérimenter et d'apprendre en ligne pour améliorer la fiabilité du modèle prédictif. Il peut aussi servir à illustrer les différences qualitatives entre plusieurs contextes opérationnels.

La figure 2.9 indique quelles sont les conséquences de la première réduction de l'espace de conception. La figure (a) représente un cas de conception non régulé entre plusieurs équipements de même flexibilité. La radio est soumise à un cadre de régulation beaucoup plus stricte dans la figure (b). Des configurations peuvent aussi avoir été écartées pour tenir compte de la flexibilité limitée de certains partenaires de la communication.

La figure 2.10 représente l'impact cumulé des spécifications de QoS et de l'environnement radio sur le partitionnement de l'espace de conception. La figure (a) correspond à un contexte plutôt favorable alors que le moteur cognitif est susceptible de consommer plus de ressources pour le contexte associé à la figure (b). Les chances de proposer une configuration économe en ressources augmentent en effet avec le nombre de configurations compatibles donc elles augmentent lorsque la radio évolue dans un canal de propagation favorable et/ou lorsque les exigences de QoS sont basses.

La figure 2.11 permet d'illustrer l'influence de l'expérience du moteur cognitif vis-à-vis du problème traité. Le moteur cognitif a une bonne compréhension de son contexte dans la figure (a) (peu d'erreurs de classification). Cette compréhension est le résultat d'expérimentations passées réalisées sur des problèmes similaires. Les erreurs de classification restantes s'expliquent de deux manières. Les expérimentations passées peuvent ne pas avoir encore abouti sur un modèle prédictif parfait. Le moteur cognitif peut aussi avoir généralisé le modèle de façon trop brutale en supposant à tort que le problème traité est similaire à un problème déjà résolu. Le modèle prédictif associé à la figure (a) permet déjà une prise de décision fiable et proche de l'optimale. L'expérimentation présente par contre plus de dangers car le moteur cognitif a de plus grandes chances de dégrader temporairement la QoS. Le moteur cognitif n'a pas la même expérience dans la figure (b). Les risques d'expérimentation seront donc plus faibles et le moteur cognitif dispose d'une marge de progression importante.

Dans ce chapitre, nous avons analysé les réflexions actuelles liées au développement d'un comportement intelligent pour une radio. Nous sommes ensuite partis de cette étude pour établir des spécifications guidant la conception de notre moteur cognitif. Nous avons enfin donné une première idée de notre approche en limitant la présentation à un niveau conceptuel. Nous exposons les détails techniques et méthodologiques pour sa mise en oeuvre au chapitre suivant.

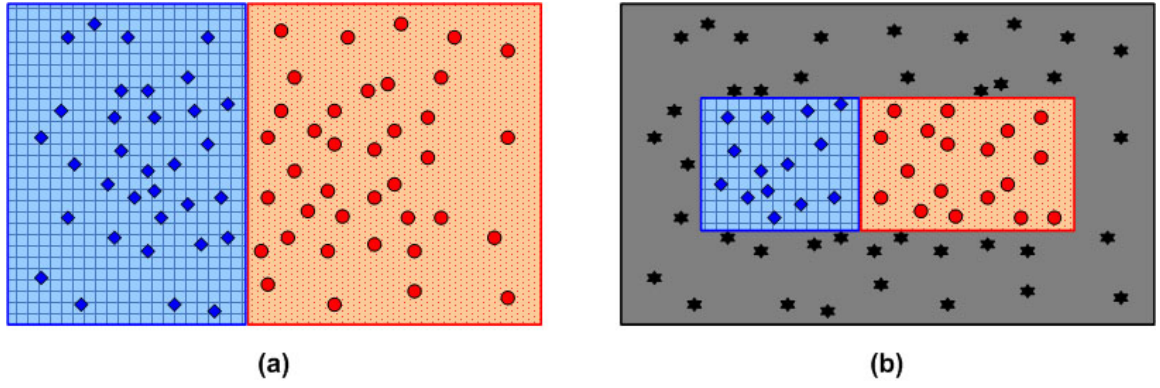


FIGURE 2.9 – Partitionnement de l'espace de conception sans (a) ou avec (b) opération de réduction préalable liée au cadre de régulation et aux contraintes matérielles des objets communicants

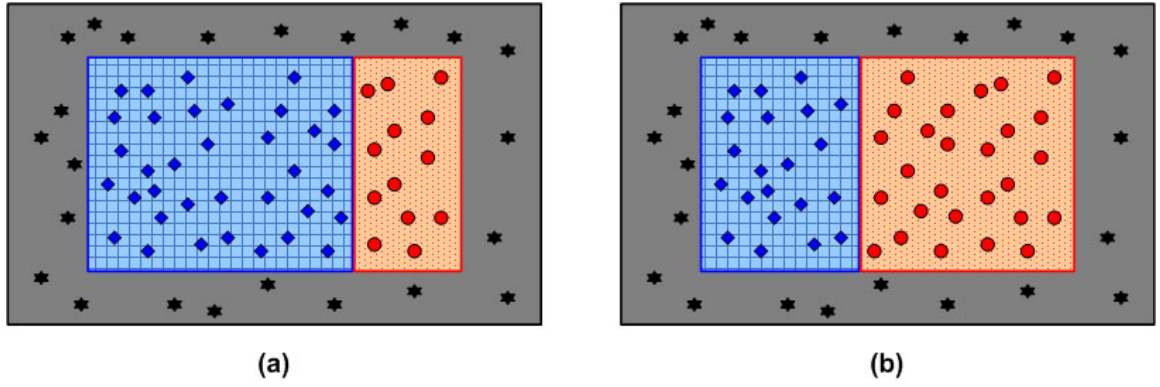


FIGURE 2.10 – Partitionnement de l'espace de conception pour un exemple de contexte favorable (a) ou défavorable (b)

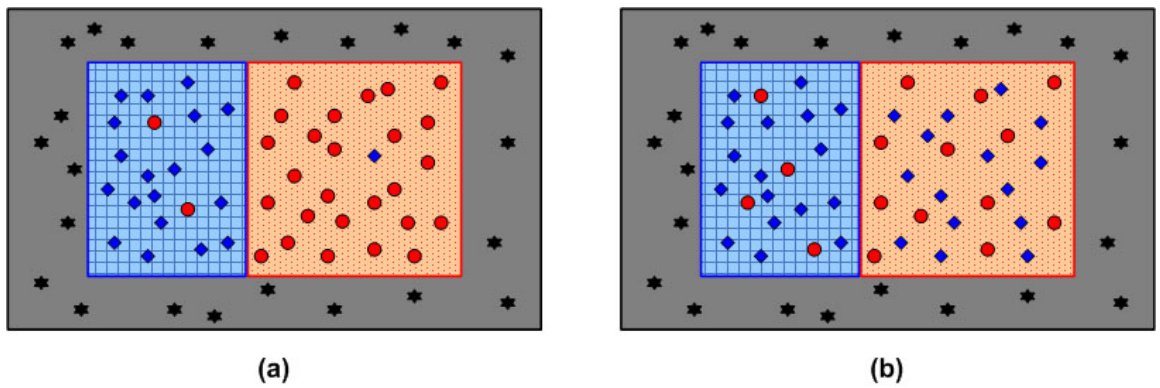


FIGURE 2.11 – Partitionnement de l'espace de conception pour un moteur cognitif ayant accumulé plus (a) ou moins (b) d'expérience vis-à-vis du problème traité

3

Démarche de conception du moteur cognitif

Sommaire

3.1	Scénario considéré pour l'évaluation du moteur cognitif	34
3.2	Modélisation du problème de prise de décision autonome	37
3.2.1	Hierarchisation des objectifs et étapes de résolution du problème	37
3.2.2	Introduction de relations d'ordre entre les alternatives	39
3.2.3	Formalisation de la prise de décision comme un problème de classification	41
3.2.4	Avantages du modèle proposé	46
3.3	Codification de l'expertise	48
3.4	Cadre d'apprentissage	52
3.4.1	Discussion sur les différentes formes d'apprentissage	53
3.4.2	Buts de l'apprentissage	55
3.5	Techniques d'intelligence artificielle choisies	56
3.5.1	Pour la prédiction des performances	57
3.5.2	Pour la gestion du dilemme exploration/exploitation	59
3.5.3	Pour la sélection de la configuration la plus adaptée	62

Le comportement cognitif d'une radio se manifeste à travers une sensibilité au contexte opérationnel et une prise de décision experte amplifiée par des capacités d'apprentissage. Le moteur cognitif part d'une observation intelligente de son contexte pour identifier le problème de conception instantané puis le résoudre. Le processus de décision est guidé par un ensemble d'objectifs (*e.g.* réduire la consommation d'énergie, maximiser le débit, minimiser les interférences) et de contraintes (*e.g.* régions interdites du spectre, puissance d'émission maximale, fiabilité minimum du lien radio) traduisant la notion d'optimalité pour le contexte identifié. Le moteur cognitif explore l'espace de conception à la recherche de la configuration la plus adaptée. Il s'appuie notamment sur ses capacités de prédiction pour estimer l'impact de l'environnement radio sur les performances des configurations autorisées. Il peut aussi décider de proposer une expérience de conception afin de vérifier l'intégrité de ses connaissances dans le but de se rapprocher du comportement optimal.

Le chapitre précédent a présenté les bases conceptuelles de nos travaux. Dans ce chapitre, nous exposons les solutions techniques et méthodologiques identifiées pour mettre en pratique les concepts proposés. Nous détaillons dans un premier temps le scénario de reconfiguration général adopté puis nous présentons notre modélisation originale du problème de décision. Nous

précisons ensuite comment l'expertise, une fois codifiée, peut permettre de limiter les risques d'expérimentation tout en accélérant l'apprentissage. Nous discutons après sur les conditions d'apprentissage et sur les bénéfices que l'on peut en attendre. Nous justifions enfin nos choix méthodologiques vis-à-vis des techniques d'intelligence artificielle sélectionnées.

3.1 Scénario considéré pour l'évaluation du moteur cognitif

Le scénario adopté est illustré à la figure 3.1. Le moteur cognitif supervise la reconfiguration du lien radio en sélectionnant les blocs de traitement du signal dynamiquement en fonction de la difficulté du canal radio et des objectifs imposés.

Le moteur cognitif prend ses décisions par rapport à trois objectifs de conception spécifiés en fonction du contexte. Les radios doivent maintenir en priorité une QoS donnée en garantissant un taux d'erreurs binaires (BER : Bit Error Rate) en dessous d'une valeur maximale. Le moteur cognitif cherche ensuite à maximiser le débit de données. Il peut aussi avoir à préserver l'énergie de la batterie en minimisant la complexité algorithmique des algorithmes de traitement du signal appliqués.

Dans ce scénario, les équipements ont une flexibilité limitée mais suffisante pour déjà offrir une large gamme de compromis possibles entre les objectifs. Ils peuvent modifier leur modulation et leur code correcteur d'erreurs en fonction de la qualité du lien radio. Le récepteur peut aussi changer d'égaliseur en fonction de la sélectivité du canal. Ce second cas d'adaptation est très souvent suggéré dans l'état de l'art (*e.g.* [19]) mais il n'a pas encore été traité selon une approche cognitive à notre connaissance. Il présente en effet certaines difficultés pour les solutions reposant sur des modèles analytiques pour la prise de décision (*e.g.* approche exploratoire). Les expressions analytiques utilisées introduisent un biais de modélisation dans le calcul des performances des configurations. Le moteur cognitif peut alors sélectionner une configuration *a priori* satisfaisante pour se rendre compte qu'elle se comporte beaucoup moins bien en réalité. Les expressions analytiques sont simples à utiliser pour un canal à bruit blanc additif gaussien (AWGN : Additive White Gaussian Noise) [4] mais elles sont beaucoup plus difficiles à obtenir pour des modèles de canaux plus complexes [107]. De plus, les expressions dérivées sont souvent exigeantes en ressources algorithmiques ce qui pénalise la batterie de l'équipement. Cet inconvénient est amplifié pour l'approche exploratoire puisque le moteur cognitif évalue plusieurs solutions en parallèle sur plusieurs générations.

Ce scénario de conception cognitive du lien radio nous apparaît donc intéressant pour mettre en valeur notre approche (basée sur une modélisation boîte noire) et la valider à l'aide de résultats expérimentaux.

Le moteur cognitif est testé sur sa capacité à résoudre un problème de conception donné. Chaque problème est caractérisé par :

- un état particulier du canal radio,
- un ensemble de politiques guidant et contraignant les choix du moteur cognitif.

Les politiques de conception sont mises au point par le moteur d'inférence lors de la phase d'analyse du contexte. De manière générale, nous distinguons deux types de politiques. Les politiques de restriction limitent l'espace de conception en imposant des contraintes à vérifier (cadre de régulation, capacités des équipements, exigences de QoS). Les politiques d'optimisation définissent des préférences entre les objectifs afin de quantifier le compromis recherché (*e.g.* préférer les configurations simples pour préserver l'énergie de la batterie).

Les radios profitent de la période d'initialisation de la communication pour réaliser des mesures sur leur environnement radio (*e.g.* SNR, profil du canal). Elles communiquent au départ avec

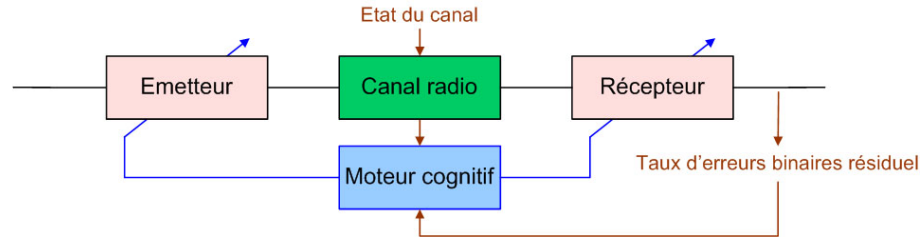


FIGURE 3.1 – Scénario de communication considéré pour tester le moteur cognitif

une configuration prédéfinie (généralement robuste). Ce lien radio initial leur permet d'échanger des informations pour la mise au point des objectifs opérationnels. Ces informations sont exprimées dans un langage standard afin de favoriser leur traitement. Elles concernent le cadre de régulation, la flexibilité des équipements engagés, les préférences des utilisateurs (de l'opérateur pour une station de base) et les contraintes énergétiques des équipements portables. Le moteur d'inférence entretient une base de connaissances mémorisant les politiques de régulation diffusées, les profils matériels des équipements fréquemment rencontrés, les profils des applications disponibles ainsi qu'un modèle de l'utilisateur :

- Les politiques de régulation spécifient quelles sont les bandes accessibles ainsi que le niveau de puissance d'émission maximal autorisé.
- Le profil d'un équipement indique quels sont les paramètres sur lesquels le moteur cognitif peut jouer.
- Le profil de l'application précise quelles sont les performances minimales attendues pour délivrer correctement le service. La notion de performance prend un sens différent pour chaque application. Elle se traduit par un ensemble de valeurs cibles pour les indicateurs de performance jugés clés pour le service considéré (*e.g.* garantir un BER en dessous de 10^{-3}). Le profil peut proposer plusieurs niveaux de QoS en fonction des codes source applicables (*cf.* sec. 2.3). Il peut aussi donner des consignes d'optimisation pour mieux guider le processus de conception (*e.g.* augmenter le débit de transmission pour une application de transfert de fichiers).
- Le modèle de l'utilisateur mémorise ses préférences qualitatives vis-à-vis des critères le concernant (*e.g.* QoS, autonomie énergétique, coût). Un professionnel recherche, par exemple, à obtenir la meilleure QoS pour travailler efficacement malgré un coût élevé (*e.g.* pour louer une large bande fréquentielle [108]). Il sera par contre plus sensible au coût du service dans un contexte non professionnel.

La phase d'analyse du contexte est réalisée en grande partie selon une approche experte : le moteur cognitif s'appuie sur la logique événementielle du moteur d'inférence pour identifier les politiques de conception s'appliquant au contexte. Le moteur cognitif pourrait néanmoins profiter de modèles prédictifs pour caractériser plus précisément son contexte et ainsi dériver des objectifs plus en adéquation avec les mesures récoltées. Des études plus poussées sont cependant nécessaires dans cette direction.⁸ A titre d'exemple, Koutsorodi *et al.* [109] utilisent des réseaux bayésiens pour modéliser les préférences de l'utilisateur.

Notre scénario de conception implique une communication entre deux équipements de même flexibilité avec un accès traditionnel au spectre (la première opération de réduction n'est donc pas simulée). Les deux radios communiquent sur une bande fréquentielle dédiée, de plus ou moins

8. Nos travaux peuvent aussi servir à la phase d'analyse du contexte. Au chapitre 6, nous proposons une méthode pour prédire la QoS que l'on peut attendre d'un canal radio donné.

bonne qualité. Un faible SNR en réception peut se traduire par des conditions de propagation défavorables et/ou par la présence d'un interférent. Le moteur cognitif cherche alors à adapter les paramètres des radios afin de maintenir la qualité du service sans avoir à changer de fréquence. Le système de communication gagne ainsi en efficacité spectrale tout en garantissant son engagement vis-à-vis de l'utilisateur.

Le moteur cognitif est testé suivant une approche statique puisqu'il traite une série de problèmes de conception indépendants l'un de l'autre.⁹ Pour chaque problème, il peut décider d'expérimenter afin de vérifier ses connaissances vis-à-vis du comportement optimal. Il peut aussi décider de faire confiance à ses connaissances et ainsi choisir la configuration qu'il juge la plus adaptée. Les performances des configurations sélectionnées sont évaluées à partir d'une modélisation analytique du système de communication décrite en annexe A. Le moteur cognitif se base sur les valeurs de BER retournées par le simulateur pour construire son plan d'expérience dynamiquement. Dès qu'une décision finale est prise, l'environnement de simulation passe au problème de conception suivant. Le moteur cognitif garde la trace de toutes les expériences de conception vécues grâce à ses capacités d'apprentissage. Il se montre donc de plus en plus efficace au fur et à mesure que la simulation progresse.

Grâce à son modèle prédictif, notre moteur cognitif est capable d'identifier lorsque le problème n'a pas de solutions parmi les configurations disponibles (*i.e.* l'ensemble des configurations compatibles est vide). Le moteur cognitif peut se comporter de plusieurs manières dans ce cas.¹⁰ En pratique, nous considérons que le problème de conception est résolu si le moteur cognitif identifie ce cas défavorable. Des études plus poussées (et un environnement de simulation plus complexe) seraient nécessaires pour gérer et analyser le comportement du moteur cognitif dans ce cas.

Les problèmes de conception sont générés aléatoirement par l'environnement de simulation. Ils sont définis par :

- Une valeur du SNR entre 0 et 20 dB,
- Un modèle de canal,
- Une valeur seuil pour le BER comprise entre 10^{-6} et 10^{-2} ,
- Une politique d'optimisation parmi les trois définies au tableau 3.1.

La politique d'optimisation est spécifiée sous la forme de poids traduisant l'importance relative

9. L'approche statique est pratique pour des questions d'implémentation. Un moteur cognitif doit cependant savoir opérer dans un environnement dynamique. Notre moteur cognitif est conçu pour adapter le comportement de la radio au rythme des variations lentes du canal (pertes de propagation, effet de masque). Les variations rapides (évanouissements) sont prises en charge par les techniques de traitement du signal sélectionnées (e.g. codage canal, égalisation, techniques de diversité). Le moteur cognitif récupère les mesures/estimations de la radio et il analyse leurs tendances afin d'identifier des situations particulières auxquelles il réagit. Il est capable, par exemple, de reconnaître lorsque l'utilisateur se trouve en condition de forte mobilité (fréquence Doppler élevée) et/ou lorsqu'il s'éloigne de la station de base (diminution du SNR). Il emploiera des algorithmes adaptatifs dans le premier cas (e.g. égaliseur en mode poursuite) et il compensera la dégradation de QoS en choisissant une configuration plus robuste dans le second cas. Nous abordons la gestion d'un environnement dynamique au chapitre 6. Nous y proposons notamment un processus de surveillance permettant au moteur cognitif de suivre l'évolution du contexte opérationnel.

10. Nous donnons ici quelques suggestions d'adaptation possible. Si la dégradation du lien radio résulte de la présence d'un interférent, le moteur cognitif peut décider de changer de bande fréquentielle. Si le spectre est encombré, il doit rester dans cette bande et faire au mieux avec les conditions de propagation rencontrées. Il peut alors décider de maintenir la configuration la plus robuste afin de limiter la dégradation des performances au maximum. Il peut aussi choisir de réviser les contraintes de QoS en modifiant le code source appliqué (e.g. passer d'une vidéo haute définition à une vidéo de qualité réduite). Pour des applications non temps-réel, il peut enfin décider d'attendre de meilleures conditions de propagation afin de ne pas gaspiller inutilement l'énergie de sa batterie.

entre les deux objectifs secondaires considérés (*i.e.* maximiser le débit de données et réduire la consommation d'énergie).

	Mode	Débit de données	Consommation d'énergie
OP_1	Multimédia	1	0
OP_2	Basse consommation	0	1
OP_3	Compromis	0.5	0.5

TABLE 3.1 – Politiques d'optimisation considérées (OP_{1-3})

L'environnement de simulation a été développé en C++. Le simulateur est configuré à l'aide d'un fichier XML stockant les paramètres de l'environnement de test (modèles de canaux utilisés, options de configuration disponibles) et du moteur cognitif. Le fichier XML est analysé en début de simulation afin de construire les différents objets logiciels impliqués dans l'étude de cas considérée. Le comportement idéal est identifié à l'avance en utilisant les expressions analytiques pour connaître les performances exactes de chaque configuration. Le moteur cognitif est ensuite testé sur chaque problème de conception et ses décisions sont mémorisées pour pouvoir les analyser en fin de simulation. Les résultats sont ensuite stockés dans un fichier puis affichés sous Matlab.

Nous concluons cette discussion sur le scénario adopté par une dernière remarque sur le déroulement de la thèse. Nous avons développé nos mécanismes cognitifs en innovant progressivement sur des études de cas de plus en plus complexes. Le degré de flexibilité des radios augmente et nous passons d'un canal AWGN vers un modèle de canal à évanouissements sélectifs. Nous avons rencontré deux étapes principales lors du développement des mécanismes. Nous avons d'abord développé nos mécanismes pour un espace de conception totalement ordonné (Chap. 4) puis nous nous sommes rendus compte de leurs limites pour des scénarios plus complexes. Nous avons alors proposé une extension de ces mécanismes afin de valider la généralité de notre approche (Chap. 5).

3.2 Modélisation du problème de prise de décision autonome

La radio est gouvernée par des objectifs et des contraintes de différente nature. Nous mettons en valeur leurs différences dans un premier temps puis nous partons de cette analyse pour définir deux échelles permettant au moteur cognitif de caractériser l'espace de conception. Nous détaillons enfin notre modélisation du problème de prise de décision et nous présentons ses avantages en matière de raisonnement et d'apprentissage en ligne.

3.2.1 Hiérarchisation des objectifs et étapes de résolution du problème

Notre moteur cognitif procède en trois étapes pour résoudre le problème de conception posé par son contexte opérationnel (*cf.* figure 3.2). Lors de la première étape, le moteur cognitif élimine de son espace de conception toutes les configurations incompatibles avec le cadre de régulation en vigueur, avec les contraintes matérielles des objets communicants ainsi qu'avec les exigences éventuelles de l'opérateur. Lors de la seconde étape, il s'appuie sur son modèle prédictif pour écarter toutes les configurations qu'il juge incapables de délivrer le service requis. La décision finale est prise lors de la troisième et dernière étape. Le moteur cognitif explore l'espace de conception doublement réduit pour sélectionner la configuration la plus en phase avec la politique d'optimisation dérivée.

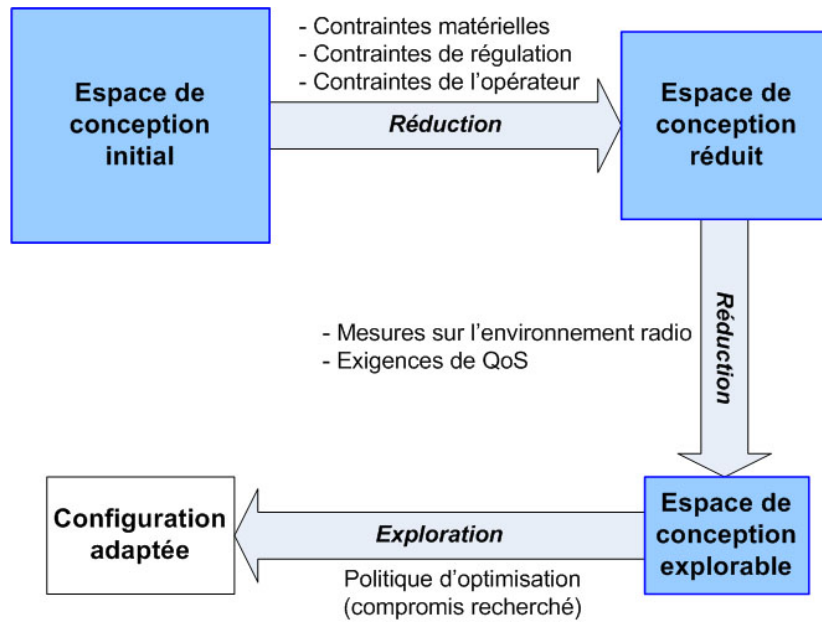


FIGURE 3.2 – Étapes de la résolution du problème de conception

Cette structuration des opérations résulte d'une hiérarchisation des objectifs sur trois niveaux. Nous faisons en effet la distinction entre :

- *Les contraintes de faisabilité* : Les contraintes de faisabilité correspondent à la fois aux contraintes matérielles (liées à la flexibilité et à l'agilité des équipements communicants) et aux contraintes de coexistence (imposées par le cadre de régulation en vigueur). Ces contraintes sont prises en compte lors de la première opération de réduction de l'espace de conception.
- *Les contraintes de performance* : Les contraintes de performance sont définies en fonction du service sélectionné. Elles se manifestent sous la forme de bornes supérieures (*e.g.* BER, latence) ou inférieures (*e.g.* débit d'informations) représentant les performances minimum requises pour délivrer le service correctement. Le moteur cognitif s'appuie alors sur une analyse prédictive des conditions de propagation pour identifier toutes les configurations susceptibles de fournir un lien radio suffisamment fiable.
- *Les préférences d'optimisation* : Les préférences d'optimisation guident la recherche d'une solution préférentielle parmi l'ensemble des configurations compatibles avec les contraintes. Le moteur cognitif distingue les configurations restantes en fonction de leurs caractéristiques intrinsèques et il sélectionne la configuration affichant le meilleur compromis vis-à-vis des préférences énoncées.

La distinction entre les deux dernières étapes résulte de l'observation suivante : l'environnement radio a une influence seulement sur la réalisation de certains objectifs. Le moteur cognitif peut évaluer les autres objectifs en analysant chaque configuration indépendamment du canal. Le tableau 3.2 distingue les objectifs présentés à la section 2.3 en fonction de leur dépendance éventuelle à l'environnement radio. Dans ce tableau, nous faisons notamment la différence entre le débit d'information (quantité d'information utile correctement reçue par unité de temps) et le débit de transmission (quantité d'information utile émise par unité de temps). Le premier objectif est influencé à la fois par la qualité du canal radio et par la forme d'onde utilisée alors que le second ne dépend que des paramètres du modulateur.

Objectifs dépendants	Objectifs indépendants
<ul style="list-style-type: none"> - <i>Minimiser le taux d'erreurs</i> - Maximiser le rapport signal/interférence plus bruit - Minimiser le délai de communication - Maximiser le débit d'informations 	<ul style="list-style-type: none"> - <i>Maximiser le débit de transmission</i> - <i>Minimiser la complexité algorithmique des traitements appliqués</i> - Minimiser la puissance consommée - Minimiser les interférences engendrées - Maximiser la bande spectrale occupée - Maximiser l'efficacité spectrale

TABLE 3.2 – Objectifs dépendants ou indépendants de l'environnement radio.
Les objectifs considérés dans la thèse figurent en italique.

Les objectifs considérés dans notre scénario sont mis en valeur en italique dans le tableau. Le moteur cognitif prend ses décisions en fonction d'une contrainte de performance et de deux objectifs d'optimisation. La contrainte de performance est spécifiée à travers la politique de restriction (valeur maximale pour le BER). La politique d'optimisation précise quelles sont les préférences relatives entre les deux objectifs secondaires.

3.2.2 Introduction de relations d'ordre entre les alternatives

La première échelle, l'*échelle de performance* (PS : Performance Scale), ordonne les configurations en fonction de leur robustesse face aux contraintes de QoS (le nombre d'erreurs de transmission dans notre cas).

En règle générale, cette échelle dépend de la difficulté du canal radio. Il est possible néanmoins de s'appuyer sur l'expertise des ingénieurs radio pour introduire des relations d'ordre au moins partielles entre les configurations. Il s'avère, par exemple, que la modulation BPSK est moins affectée par des erreurs de transmission que des modulations d'ordre plus élevé. Cette meilleure tolérance au bruit résulte d'une plus grande distance de séparation entre les symboles de la constellation. L'ingénieur radio peut donc se baser sur cette distance de séparation pour ordonner toutes les modulations disponibles (M-PSK et M-QAM).¹¹

Lorsque l'on considère la configuration dans son intégralité, ce "bon sens" expert peut être complété par des connaissances acquises au travers de simulations hors-ligne. Les simulations sont réalisées dans un premier temps avant la commercialisation du terminal. Les performances des configurations sont comparées, par exemple, sur des modèles de canaux typiques. Le classement des configurations peut aussi être mis à jour grâce aux informations récupérées par les terminaux lorsqu'ils expérimentent. Le moteur cognitif peut alors comparer les performances des configurations expérimentées et en déduire leur robustesse relative vis-à-vis du canal radio considéré. Les informations récoltées par chaque radio peuvent aussi être remontées au niveau des stations de base afin de profiter de leur vision globale et de leurs capacités de calcul. Les stations de base analysent les informations reçues en tâche de fond afin d'extraire les relations d'ordre persistantes associées aux conditions de propagation locales. Elles les communiquent aux terminaux lors du processus de connexion afin de mieux orienter leurs expérimentations. Cet aspect est important, notamment pour de larges espaces de conception qui seraient trop long à explorer pour une radio seule. Chaque radio participe alors à la structuration progressive de l'espace de conception pour l'environnement radio associé à la cellule.

11. La distance de séparation est calculée à partir de considérations géométriques sur la forme de la constellation. Des expressions sont disponibles, par exemple, dans [70].

La seconde échelle, l'*échelle d'optimalité* (OS : Optimality Scale) range les configurations en fonction de leur optimalité potentielle¹² vis-à-vis des objectifs d'optimisation.

L'échelle d'optimalité est établie à l'aide d'un système de notation évaluant les alternatives disponibles à chaque fois qu'une nouvelle politique d'optimisation est appliquée. Le système de notation examine les paramètres des configurations pour leur attribuer des notes en fonction des préférences relatives entre les objectifs secondaires. Ces notes traduisent alors l'adéquation de chaque alternative au problème.

Les exemples suivants permettent de mieux saisir l'intérêt des objectifs secondaires pour la sélection d'une configuration particulièrement adaptée.

Le débit d'information est toujours plus faible que le débit de transmission suite aux distorsions introduites par le canal. La différence de performance peut cependant être compensée par des algorithmes de traitement du signal. Un égaliseur performant peut ainsi éliminer les interférences inter-symboles et réduire le nombre d'erreurs de transmission à un nombre suffisamment faible pour permettre une récupération fiable des informations transmises. Le système de communication peut alors se suffire d'un code correcteur simple (voire aucun code) afin de limiter la redondance des données et ainsi augmenter le débit d'information utile. De la même manière, une modulation sera plus attrayante au niveau du débit si elle offre une bonne efficacité spectrale.

Lorsqu'il est préférable de préserver l'autonomie de la batterie, le moteur cognitif privilégie les configurations les plus simples parmi celles vérifiant les contraintes de performance. Il connaît l'impact de chaque algorithme sur la consommation d'énergie donc il est à même de choisir la configuration assurant le meilleur compromis fiabilité/complexité.

Pour diminuer le délai de la communication, le moteur cognitif peut tenir compte de la rapidité de traitement des algorithmes lors de sa prise de décision. Prenons l'exemple de l'opération d'entrelacement/désentrelacement. Les erreurs introduites par le canal ne se font généralement pas de manière ponctuelle. Elles affectent alors un certain nombre de symboles consécutifs. L'entrelacement permet de disperser les rafales d'erreurs induites par le canal afin d'apporter une diversité temporelle profitable à la qualité du lien radio. Les blocs de données sont réorganisés suivant un mapping prédéfini avant transmission. En réception, le signal est désentrelacé et les rafales d'erreurs (affectant des symboles adjacents) sont éparpillées, chaque bloc se retrouvant avec peu ou pas d'erreurs. Il devient alors possible de détecter et/ou de corriger ces erreurs par des techniques de codage efficaces. L'entrelaceur a le désavantage par contre d'augmenter la latence de la transmission suite à la recombinaison des trames dans le bon ordre. Cet effet peut être indésirable pour des communications temps réel. De plus, l'efficacité de l'entrelacement est considérablement réduite si le canal se retrouve dans un état où la latence maximale tolérée est plus petite que la durée d'un évanouissement profond [1]. Le moteur cognitif n'a donc pas toujours intérêt à mettre en place un entrelacement des données, surtout s'il a la possibilité de s'en affranchir pour délivrer le service correctement. Ce raisonnement s'applique aussi au reste de la chaîne : la rapidité de traitement de chaque algorithme (en nombre de bits traités par seconde) est mesurée avant production sur la plateforme commercialisée en supposant un canal parfait. Ces informations sont ensuite stockées dans le profil de l'équipement pour que le moteur cognitif sache quelles configurations privilégier pour les communications temps réel.

12. La configuration au sommet de l'échelle n'est effectivement optimale que si elle vérifie les contraintes de performance.

3.2.3 Formalisation de la prise de décision comme un problème de classification

Les deux échelles proposées guident le moteur cognitif dans sa recherche de la configuration la plus adaptée. L'échelle de performance lui permet de raisonner sur la difficulté du problème de conception posé alors que l'échelle d'optimalité exprime la satisfaction intrinsèque de chaque configuration vis-à-vis du compromis recherché.

Le moteur cognitif commence par une analyse prédictive du problème de conception afin d'estimer les performances des configurations relativement à la contrainte de QoS imposée. Cette analyse est réalisée en deux temps. Le moteur cognitif cherche d'abord à identifier l'échelle de performance associée à l'environnement radio modélisé. L'échelle de performance est connue à l'avance pour un espace de conception totalement ordonné alors qu'elle doit être prédite si le moteur cognitif ne dispose que de relations d'ordre partielles. Le moteur cognitif s'appuie ensuite sur cette échelle pour déterminer quelles sont les configurations susceptibles de délivrer le service correctement.

La figure 3.3 rappelle comment interpréter l'échelle de performance pour un espace de conception impliquant six configurations C_1 à C_6 . L'alternative C_0 est un cas spécial activé si le problème s'avère impossible compte tenu des configurations disponibles. L'échelle de performance permet de structurer localement l'espace de conception. Le moteur cognitif sait, par exemple, que les configurations $\{C_i\}_{i \leq 4}$ sont suffisamment performantes si la configuration C_4 arrive déjà à vérifier la contrainte de QoS. Le raisonnement peut être poussé plus loin. Le moteur cognitif connaît en effet toutes les configurations compatibles avec le service (*i.e.* C_1 à C_4) s'il se rend compte que C_5 occasionne un nombre excessif d'erreurs de transmission. Le problème de prédiction des performances est donc résolu si le moteur cognitif mémorise cette association entre le problème posé et la configuration compatible la moins robuste (*i.e.* C_4).

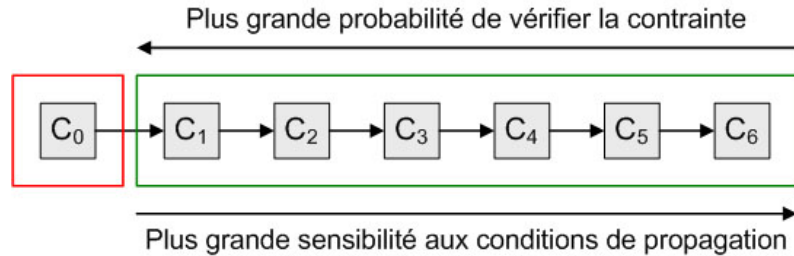


FIGURE 3.3 – Interprétation du classement des configurations selon l'échelle de performance

Cette affirmation n'est valide qu'à condition de bénéficier de connaissances fiables sur les relations d'ordre locales entre les alternatives. Cette condition est acquise pour un espace de conception totalement ordonné. Par contre, elle n'est pas toujours vérifiée pour un espace de conception partiellement ordonné. Le moteur cognitif a alors recours à l'expérimentation pour réduire les erreurs de classement au maximum.

Nous avons modélisé la tâche de prédiction des performances comme un problème de classification. Le moteur cognitif cherche à catégoriser chaque problème de conception vers un ensemble de configurations compatibles. Il analyse ses expériences afin de construire et de mettre à jour une carte de performance qui associe des problèmes similaires avec un ensemble de configurations capables de les résoudre.

Nous proposons deux exemples afin de mieux appréhender le problème de prédiction tel qu'il est représenté par notre moteur cognitif. Nous présentons ensuite la procédure de décision

aboutissant à la sélection d'une configuration adaptée. Nous indiquons enfin quel est l'impact du modèle prédictif sur la qualité des décisions.

Prédiction des performances pour un cas de conception totalement ordonné

La figure 3.4 représente un cas de conception totalement ordonné. Dans cette figure, nous avons reproduit les courbes de BER en fonction du SNR pour six configurations différentes dans un canal AWGN. L'environnement de simulation est configuré pour générer des problèmes (*i.e.* une valeur de SNR, une valeur de BER cible) uniquement à l'intérieur du cadre noir représentant les frontières considérées par notre scénario. Pour cette portion d'espace, les configurations sont totalement ordonnées. En effet, nous avons toujours $C_1 \succ C_2 \succ C_3 \succ C_4 \succ C_5 \succ C_6$, avec $C_i \succ C_j$ indiquant que C_i est plus robuste que C_j vis-à-vis de la contrainte considérée.

Cet ordre s'observe facilement à partir de la position relative des courbes de performance. La robustesse des configurations disponibles peut alors s'interpréter de deux manières. Elle peut s'exprimer en termes de sensibilité aux conditions propagation pour un BER donné. Par exemple, la configuration C_1 peut atteindre un BER inférieur à 10^{-3} pour des SNR supérieurs à 3,7 dB alors que les autres configurations ont besoin de plus grandes valeurs de SNR : 5,95 dB pour C_2 , 6,75 dB pour C_3 , 11,4 dB pour C_4 , 13,8 dB pour C_5 et 14,7 dB pour C_6 . Elle peut aussi s'énoncer en termes de performances pour un environnement radio donné. Par exemple, pour un SNR de 14 dB, nous retrouvons l'ordre donné plus haut avec $BER(C_1) = 4,5 \cdot 10^{-65} < BER(C_2) = 8,8 \cdot 10^{-23} < BER(C_3) = 6,8 \cdot 10^{-13} < BER(C_4) = 4,3 \cdot 10^{-8} < BER(C_5) = 8,1 \cdot 10^{-4} < BER(C_6) = 2,2 \cdot 10^{-3}$. Nous remarquons cependant que cet ordre n'est réellement observé que pour des SNR plus grands que 12 dB. En pratique, nous avons décidé de ne pas tenir compte des performances relatives des configurations pour des valeurs de BER supérieures à 10^{-2} étant donné qu'un BER cible aussi élevé ne peut conduire qu'à des services très dégradés donc inintéressants. Cette tolérance aux imprécisions nous permet alors de maintenir les mêmes relations d'ordre pour l'intégralité de l'espace de conception. Des connaissances précises et exactes n'apporteraient aucune amélioration sur la qualité des décisions prises alors qu'elles viendraient complexifier les mécanismes cognitifs en encombrant inutilement la mémoire du moteur cognitif.

Dans la figure, nous avons aussi mis en valeur les différentes régions de classification. L'espace de classification apparaît plus facilement à l'aide d'exemples :

- L'exemple Ex_1 est caractérisé par un BER cible égal à 10^{-3} et un SNR de 2 dB. Pour un SNR aussi faible, aucune configuration n'est capable de limiter le nombre d'erreurs de transmission au niveau requis donc l'alternative C_0 est la plus appropriée. Le même raisonnement s'applique à l'ensemble des problèmes définis sur la zone associée à C_0 .
- L'exemple Ex_2 est caractérisé par un BER cible égal à 10^{-3} et un SNR de 14,5 dB. Ce problème de conception se trouve dans la région associée à l'alternative C_5 . Notre moteur cognitif va alors considérer que C_5 correspond à la configuration compatible la moins robuste car les configurations C_1 à C_5 engendrent un BER suffisamment faible alors que C_6 n'est pas assez performante pour délivrer le service correctement. Il aura donc une connaissance complète des configurations compatibles avec le service s'il arrive à prédire que ce problème est catégorisé vers C_5 .

Le moteur cognitif profite ainsi des connaissances expertes sur les relations d'ordre entre les alternatives pour minimiser le nombre d'informations à mémoriser et ainsi augmenter l'efficacité du modèle prédictif.

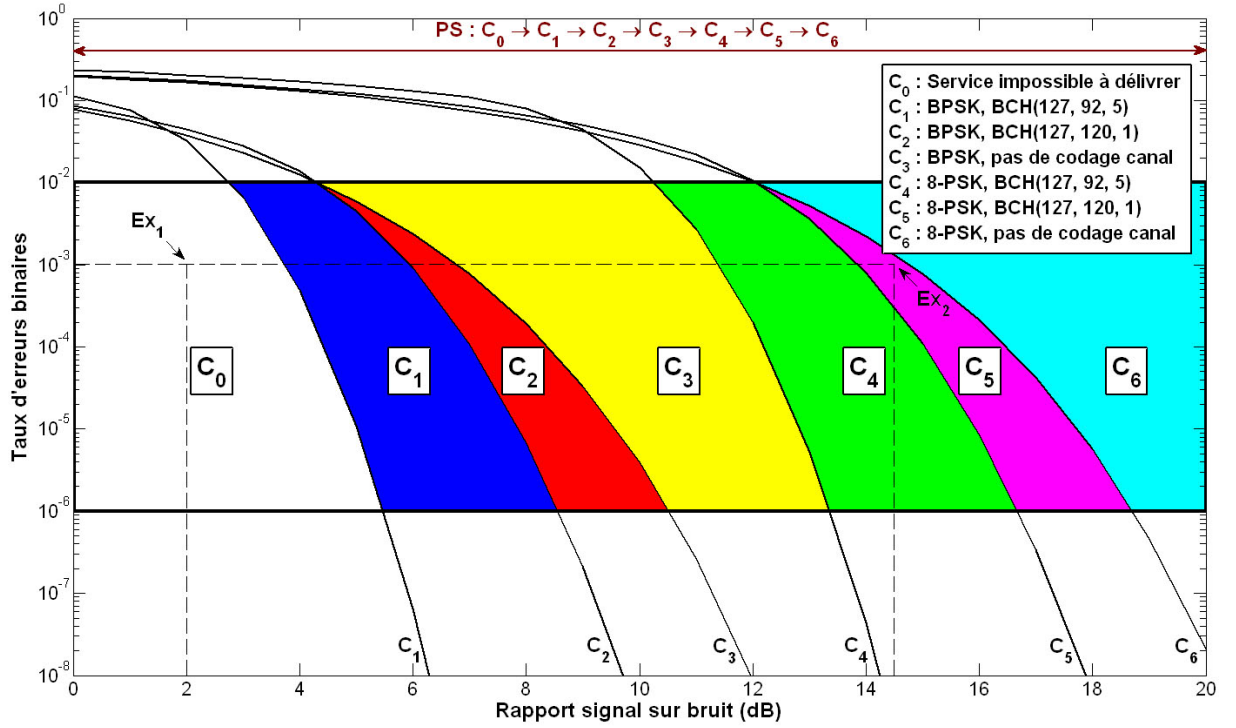


FIGURE 3.4 – Exemple d'un espace de conception totalement ordonné

C_i : Configuration associée à la région de classification, C_i : Configuration associée à la courbe de performance.

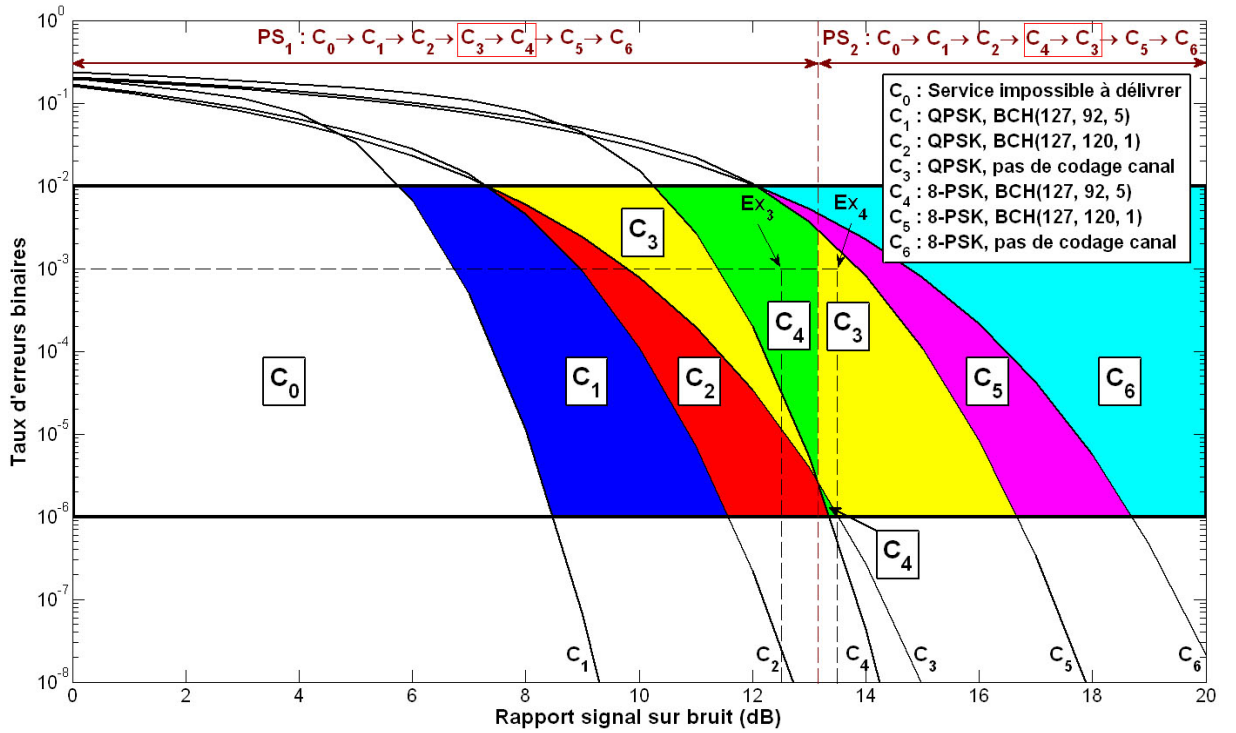


FIGURE 3.5 – Exemple d'un espace de conception partiellement ordonné

C_i : Configuration associée à la région de classification, C_i : Configuration associée à la courbe de performance.

Prédiction des performances pour un cas de conception partiellement ordonné

La figure 3.5 illustre un cas de conception lorsque le moteur cognitif ne dispose que de relations d'ordre partielles. Le scénario est semblable au précédent sauf que la modulation BPSK des trois premières configurations a été remplacée par une modulation QPSK moins robuste. Les courbes de performance de C_3 et C_4 se croisent pour un SNR de 13,2 dB donc il n'est plus possible d'ordonner ces deux configurations sans tenir compte des conditions de propagation. C_3 est plus performante pour un SNR inférieur à 13,2 dB alors que C_4 devient plus efficace passé ce seuil. Cet espace est donc caractérisé par deux échelles de performance comme l'indique la figure. Les deux échelles ne diffèrent que d'une simple permutation liée au croisement des deux courbes. Le moteur cognitif doit alors se baser sur ses capacités de prédiction pour identifier l'échelle de performance s'appliquant à son environnement radio actuel. Il apprend progressivement à classer les environnements radios en fonction des relations d'ordre qu'ils engendrent.

Le moteur cognitif s'appuie sur l'échelle de performance prédite pour déterminer quelles sont les configurations compatibles. Le principe reste le même que précédemment, le moteur cognitif associant chaque problème de conception avec sa configuration compatible la moins robuste. Le point de croisement introduit une frontière de décision supplémentaire dans l'espace de classification. Cette nouvelle frontière apparaît plus clairement en prenant un exemple de chaque côté. Pour le problème de conception associé à l'exemple Ex_3 , C_4 reste moins performante que C_3 même si elle permet d'atteindre le BER requis. Nous avons en effet $BER(C_3) = 1,2 \cdot 10^{-5} < BER(C_4) = 3 \cdot 10^{-5} < 10^{-3}$. L'échelle de performance est différente pour le SNR associé à l'exemple Ex_4 donc la configuration compatible la moins robuste correspond cette fois à C_3 au lieu de C_4 .

Sélection d'une configuration adaptée

Le moteur cognitif dispose de toutes les informations pour prendre sa décision une fois les performances prédites. Le système de notation embarqué évalue la satisfaction intrinsèque de chaque alternative vis-à-vis du compromis recherché afin d'établir l'échelle d'optimalité. Une note élevée atteste d'un comportement appréciable à l'égard des objectifs secondaires. Le moteur cognitif n'a plus qu'à sélectionner l'alternative de plus haut rang dans l'échelle d'optimalité parmi les configurations jugées compatibles avec le service. Cette procédure est illustrée à travers quatre cas présentés à la figure 3.6.

Les cas (a) et (b) s'appliquent pour tout problème de conception défini sur la zone associée à C_4 dans la figure 3.4. D'après notre modèle, les configurations C_1 à C_4 sont prédites comme performantes (SC : Satisfied Constraint) alors que C_5 et C_6 ne sont pas censés vérifier la contrainte de QoS (VC : Violated Constraint). Le système de notation sera décrit plus amplement à la section 4.2.2 mais la tendance des notes peut déjà s'expliquer qualitativement. Pour le cas (a), le moteur cognitif privilégie uniquement le débit de transmission (OP_1) or le débit de transmission augmente en passant de C_1 à C_6 (diminution de la redondance du code puis passage à une modulation d'ordre plus élevé). D'après notre modèle, C_4 présente le meilleur compromis débit/fiabilité donc c'est la configuration choisie par le moteur cognitif. Pour le cas (b), le moteur cognitif se focalise sur la consommation d'énergie (OP_2) donc il attribue de meilleurs notes aux configurations sans codage ou employant un code correcteur simple. Il choisit alors la configuration C_3 afin de s'affranchir des opérations de codage/décodage coûteuses en ressources de calcul.

Les cas (c) et (d) sont obtenus pour des problèmes de conception définis sur la petite région associée à C_4 dans la figure 3.5. Nous supposons, de plus, que le moteur cognitif se trompe dans

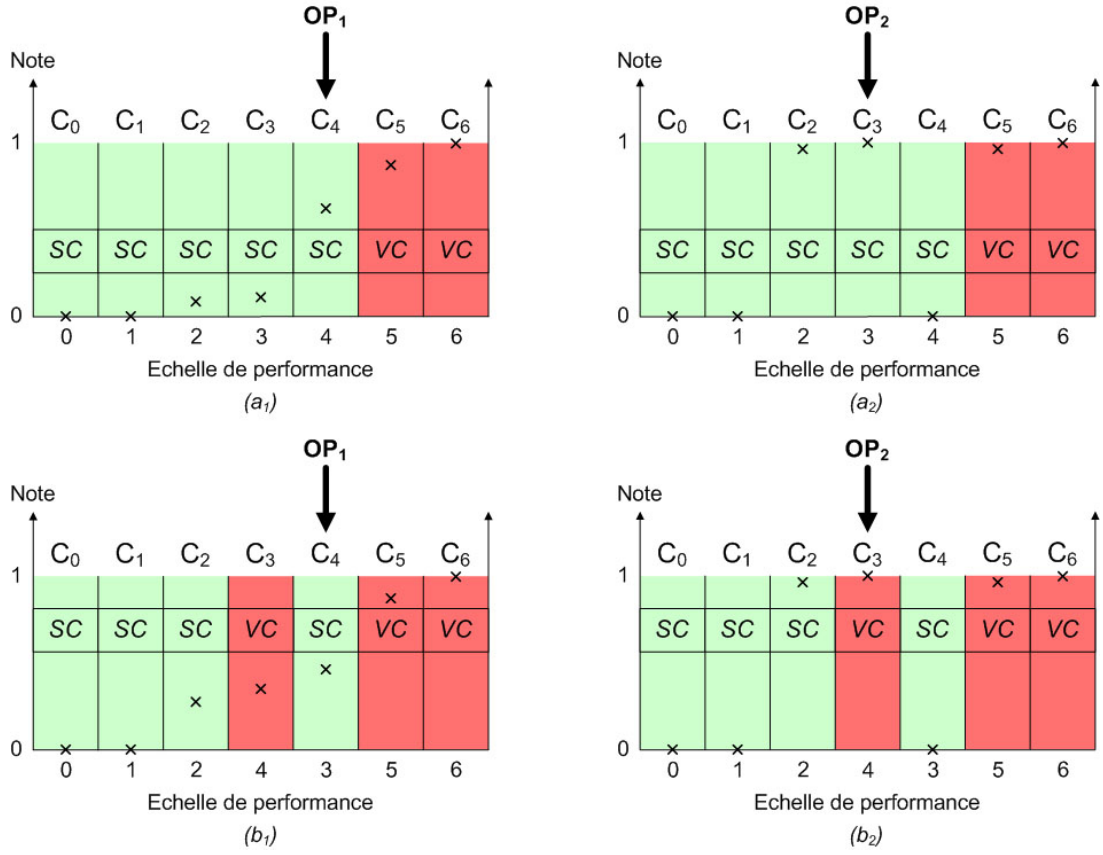


FIGURE 3.6 – Scénarios de prise de décision pour différentes politiques d'optimisation (gauche : OP_1 , droite : OP_2) avec (bas) ou sans (haut) erreurs dans l'échelle de performance
SC : Satisfied Constraint, VC : Violated Constraint

sa prédiction de l'échelle de performance. Ses connaissances actuelles lui laissent croire que C_4 est plus performante que C_3 alors que ce n'est plus le cas.

Cette erreur de prédiction n'a pas d'impact sur la prise de décision dans le cas (c) lorsque le moteur cognitif favorise le débit de transmission (OP_1). Le moteur cognitif choisit C_4 puisqu'elle apparaît comme la configuration la plus attrayante du point de vue du débit. Cette configuration vérifie les contraintes donc le moteur cognitif a pris une décision correcte voire optimale. La décision est optimale pour tous les problèmes de conception définis sur la petite région associée à C_4 et sur les régions associées à C_3 . Elle serait sous-optimale si le moteur cognitif avait été induit en erreur par son modèle prédictif estimant à tort que C_5 est trop peu performante pour délivrer le service demandé (région associée à C_5).

Le moteur cognitif prend une mauvaise décision dans le cas (d) en choisissant C_3 pour réduire la consommation d'énergie du terminal (pas de codage canal) alors que la configuration ne vérifie pas les contraintes en réalité. Il apprend alors de son erreur en retenant cette mauvaise association :

- S'il a testé C_4 au préalable, il remarque que la relation d'ordre a changé entre C_3 et C_4 et il met à jour l'échelle de performance pour le canal radio actuel et les canaux similaires (*i.e.* de SNR proche).
- S'il prend cette décision en début d'expérience, il adopte un comportement conservateur et il choisit C_2 pour tester une configuration plus robuste. Il mémorise cette association à

moins que ses expérimentations ne lui permettent d'identifier que C_4 est compatible avec la contrainte et qu'elle est plus performante que C_3 .

Influence du modèle prédictif sur la qualité des décisions prises

La configuration compatible la moins robuste est associé à un rang dans l'échelle de performance que nous appelons rang de plus haute compatibilité (HCR : Highest Compatible Rank). Le HCR est égal au nombre de configurations compatibles si l'échelle de performance est fiable. Il peut être supérieur ou inférieur à ce nombre si le modèle prédictif introduit des erreurs de classement. Un HCR plus grand augmente les chances de mauvaises décisions alors qu'un HCR plus faible a tendance à éloigner le moteur cognitif du comportement optimal. Le HCR peut aussi être égal au nombre effectif de configurations compatibles si les erreurs de classement se compensent entre les deux catégories (compatible / non compatible). La figure 3.7 illustre ces interprétations à l'aide de l'outil conceptuel proposé à la section 2.5.

La configuration optimale est encadrée pour la politique d'optimisation OP_1 et elle est au-réolée pour OP_2 . Dans le cas (a), les prédictions sont parfaites donc le moteur cognitif prend la décision optimale pour les deux politiques d'optimisation. Dans le cas (b), l'espace de conception n'est pas parfaitement structuré et il reste des configurations trop peu performantes parmi l'ensemble des configurations jugées compatibles. Cette erreur de prédiction a un impact pour OP_1 car le moteur cognitif pense choisir une configuration plus satisfaisante vis-à-vis des objectifs secondaires sans savoir qu'elle ne vérifie pas les contraintes. Elle n'en a pas pour OP_2 étant donné que la configuration optimale se trouve effectivement dans l'ensemble des configurations supposées compatibles. Dans le cas (c), la mauvaise structuration de l'espace de conception résulte en un comportement sous-optimal pour OP_1 car la configuration optimale n'est pas jugée suffisamment performante pour délivrer le service correctement. Elle n'a pas d'impact pour OP_2 là encore. Le cas (d) est soumis aux deux types d'erreurs présentées aux cas (b) et (c) même si le HCR est bien égal au nombre de configurations compatibles.

3.2.4 Avantages du modèle proposé

Notre méthode de modélisation a l'avantage de combiner efficacement la puissance de systèmes d'apprentissage avec des connaissances issues de l'expertise humaine. L'échelle d'optimalité exploite des connaissances quantitatives matérialisées par des fonctions permettant d'évaluer la satisfaction de chaque objectif secondaire en fonction des paramètres des configurations. L'échelle de performance s'appuie quant à elle sur des connaissances qualitatives sur la robustesse relative des configurations vis-à-vis des contraintes de QoS.

Nous avons vu à la section 2.4.2 qu'il n'était pas toujours possible ni judicieux de compter sur un modèle analytique pour quantifier les performances des configurations pour tous les environnements radio. Les relations d'ordre nous permettent ici d'accélérer l'apprentissage d'un modèle prédictif fiable afin d'augmenter l'autonomie décisionnelle du terminal tout en minimisant les expérimentations hasardeuses. Les connaissances sont moins précises donc elles gagnent en généralité. Elles sont aussi plus faciles à identifier que ce soit par un expert humain, un processus hors-ligne ou par le moteur cognitif lors de son opération. Enfin, elles sont plus simples à utiliser qu'un modèle analytique généralement coûteux en ressources de calcul (*e.g.* pour un canal à évanouissements sélectifs).

Le problème d'optimisation complexe a été transformé en un problème d'optimisation sous contrainte plus simple dans sa formulation. Le moteur cognitif cherche ainsi à maximiser le rang dans l'échelle d'optimalité sous la contrainte d'une identification correcte de toutes les

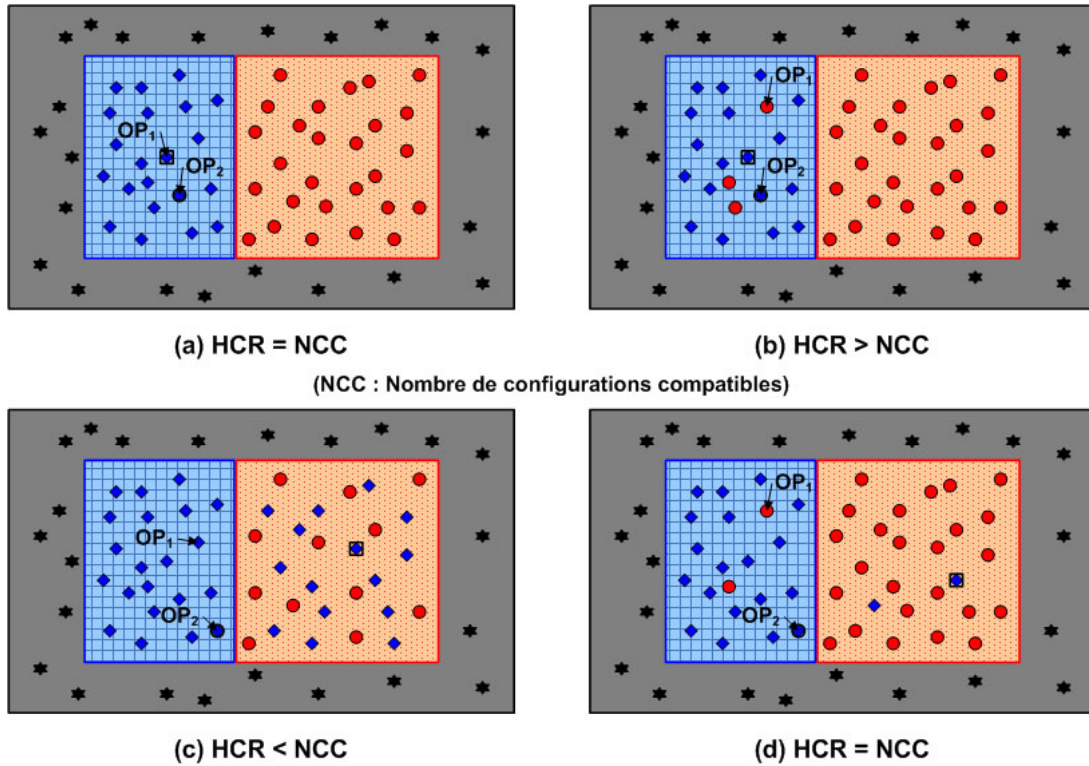


FIGURE 3.7 – Interprétation du HCR par rapport au nombre de configurations compatibles avec le service

configurations compatibles avec le service. La difficulté du problème réside alors dans la phase d'analyse prédictive : le moteur cognitif doit pouvoir estimer les performances des configurations disponibles seulement à partir de la spécification de la contrainte et d'informations statistiques sur l'environnement radio. Ce problème est résolu par la suite à l'aide de systèmes prédictifs dont la fiabilité augmente avec le temps par apprentissage.

Un modèle prédictif parfait fournit une échelle de performance fiable et une estimation correcte du HCR. Notre moteur cognitif n'a pas besoin d'arriver à ce niveau de perfection pour déjà prendre des décisions optimales. Nous avons vu en effet qu'il avait une certaine tolérance aux erreurs de prédiction. Le modèle prédictif s'avère fiable pour la prise de décision tant qu'il n'exclut pas la configuration optimale de l'ensemble des configurations prédites comme compatible avec le service. Cette tolérance aux erreurs résulte de notre approche qualitative pour la prédiction des performances. Les performances des configurations sont estimées de manière approximative : nous nous intéressons seulement à leur compatibilité avec le service sans chercher à connaître le nombre exact d'erreurs de transmission. Cette approche diffère des approches plus quantitatives comme [31] par exemple. Elle a l'avantage supplémentaire de simplifier la tâche de prédiction tout en ne diminuant pas son efficacité.

L'introduction de relations d'ordre facilite le raisonnement et l'apprentissage. L'échelle de performance structure l'espace de conception afin de guider plus efficacement son exploration. Le moteur cognitif peut ainsi raisonner en termes de progression le long de l'échelle de performance pour se rapprocher progressivement du comportement optimal. Cette logique est à l'origine du processus d'expérimentation implémenté. La configuration compatible la moins robuste est ainsi identifiée dès que la configuration de rang supérieur ne vérifie plus la contrainte. L'échelle de

performance peut ne pas toujours s'avérer fiable pour un espace de conception partiellement ordonné. Le moteur cognitif peut donc s'autoriser à tester des configurations associées à des rangs au-delà du HCR pour aller chercher des configurations potentiellement plus satisfaisantes. Il compare les performances des configurations testées et il corrige les erreurs de classement pour mettre à jour l'échelle de performance ainsi que le HCR pour le problème considéré.

Le moteur cognitif peut enfin naviguer facilement sur les deux échelles en fonction des événements mis en valeur lors de l'analyse du contexte. Il consulte son modèle prédictif lorsque l'utilisateur change d'application. Il régresse le long de l'échelle de performance pour un service plus exigeant en termes de performance alors qu'il cherche à progresser le long de cette échelle si les contraintes de QoS sont moins fortes. Il raisonne de la même manière si les conditions de propagation s'améliorent (progression) ou se détériorent (régression). Lorsque la politique d'optimisation est modifiée, le système de notation embarqué réévalue les alternatives pour réviser les équivalences entre les configurations et leurs rangs dans l'échelle d'optimalité. Le moteur cognitif n'a plus qu'à choisir la configuration la plus satisfaisante dans ce nouveau classement. Il n'a pas besoin de faire intervenir son modèle prédictif puisque l'environnement radio et les contraintes sont les mêmes.

3.3 Codification de l'expertise

Le moteur cognitif ne peut s'appuyer sur l'expertise humaine en télécommunications sans une codification préalable des connaissances dans un langage compréhensible par la machine. Notre moteur cognitif exploite deux types de connaissances :

- Les fonctions objectif paramétrées permettent de calculer la satisfaction intrinsèque des configurations vis-à-vis des objectifs secondaires. Le système de notation utilise ces fonctions pour établir l'échelle d'optimalité s'appliquant au contexte.
- Les relations d'ordre sur la robustesse relative des configurations contribuent à une structuration fiable (plus ou moins forte) de l'espace de conception au voisinage du problème considéré. Le moteur cognitif profite de ces connaissances qualitatives pour guider son exploration locale de l'espace de conception. Il peut ainsi progresser plus rapidement vers le comportement optimal tout en diminuant les risques d'expérimentation.

Dans cette section, nous nous focalisons sur l'identification, l'intégration et l'utilisation des relations d'ordre représentatives des performances relatives entre les configurations. Les fonctions objectif considérées seront présentées à la section 4.2.2 en même temps que le système de notation embarqué.

Le scénario adopté limite la reconfiguration du terminal à trois blocs de traitement du signal (codage canal, modulation, égalisation). Il est possible pour chacun de ces blocs d'ordonner les options disponibles de manière intuitive :

- Une configuration est plus robuste au bruit et aux interférences si les données sont protégées par un code introduisant une redondance élevée des informations.
- Une modulation d'ordre élevée est plus sensible aux conditions de propagation qu'une modulation moins avancée.
- Une configuration sans égaliseur se comporte généralement moins bien qu'une configuration intégrant un égaliseur linéaire ou un MLSE [110].

Nous considérons trois options de configurations par bloc ce qui fait un total de $3 \times 3 = 27$ configurations disponibles. Le tableau 3.3 présente les différentes options possibles pour chaque bloc en les classant selon leur robustesse relative et en leur associant un rang local de performance.

Rang local	Codage Canal		Modulation		Égalisation	
1	$c1$	BCH(127, 92, 5)	$m1$	BPSK	$e1$	MLSE
2	$c2$	BCH(127, 120, 1)	$m2$	QPSK	$e2$	Égaliseur linéaire
3	$c3$	Pas de codage	$m3$	8-PSK	$e3$	Pas d'égaliseur

TABLE 3.3 – Options de configuration considérées

Selon cette logique, la configuration $c1m2e1$ est toujours plus robuste que $c1m3e1$ ou $c1m2e2$. Par contre, on ne peut pas savoir *a priori* laquelle de ces deux alternatives sera la plus performante pour un problème de conception donné.

Les relations d'ordre sont mémorisées à l'aide d'un graphe acyclique orienté contenant autant d'états que de configurations disponibles. Le lien d'un état parent vers un état fils symbolise le passage d'une configuration donnée vers une configuration moins performante. La configuration au sommet du graphe correspond donc à la configuration la plus robuste parmi toutes les configurations disponibles.

La figure 3.8 représente le graphe de robustesse pour le cas de conception illustré à la figure 3.4. Les flèches noires correspondent aux relations d'ordre locales déduites lors de l'analyse par bloc. Pour générer une échelle de performance, le graphe est parcouru à l'aide de l'algorithme de tri topologique décrit à la section 4.2.1. Ce graphe, créé à partir des relations d'ordre local, n'est compatible qu'avec six échelles de performance :

- $C_0 \succ C_1 \succ C_2 \succ C_3 \succ C_4 \succ C_5 \succ C_6$
- $C_0 \succ C_1 \succ C_2 \succ C_4 \succ C_3 \succ C_5 \succ C_6$
- $C_0 \succ C_1 \succ C_2 \succ C_4 \succ C_5 \succ C_3 \succ C_6$
- $C_0 \succ C_1 \succ C_4 \succ C_5 \succ C_2 \succ C_3 \succ C_6$
- $C_0 \succ C_1 \succ C_4 \succ C_2 \succ C_5 \succ C_3 \succ C_6$
- $C_0 \succ C_1 \succ C_4 \succ C_2 \succ C_3 \succ C_5 \succ C_6$

En observant la figure 3.4, nous remarquons que C_3 est plus robuste que C_4 (position relative des courbes). Nous pouvons donc ajouter un lien entre ces deux configurations afin de réduire le nombre d'échelles compatibles à une seule (espace de conception totalement ordonné).

Le lien nouvellement créé n'est plus valable pour tous les SNR avec le cas de conception de la figure 3.5. Le moteur cognitif doit donc apprendre progressivement à identifier les échelles de performance s'appliquant aux canaux radio rencontrés. Il sait néanmoins qu'il n'a pas à tester toutes les combinaisons possibles puisque l'échelle de performance doit être compatible avec le graphe. L'échelle prédite sera donc forcément égale à l'une des six présentées plus haut.

Si l'espace de conception est bien maîtrisé, il est possible de l'ordonner complètement même en présence de relations d'ordre partielles. Il suffit dans ce cas d'ajouter des conditions aux liens de robustesse comme l'illustre la figure 3.9.

L'espace de conception se complexifie avec la flexibilité accrue des équipements. La figure 3.10 montre le graphe de robustesse obtenu lorsque l'on considère l'ensemble des configurations disponibles. Le graphe est organisé par couche de manière à regrouper les configurations en fonction de la somme de leurs rangs locaux. Une somme faible traduit une grande robustesse face à la contrainte (e.g. $c1m1e1 \rightarrow 3$) alors qu'une somme élevée indique que les configurations sont moins performantes *a priori* (e.g. $c3m3e3 \rightarrow 9$).

Les relations d'ordre partielles identifiées grâce à l'analyse par bloc permettent d'éliminer un grand nombre de combinaisons mais le nombre d'échelles possibles peut rester élevé pour

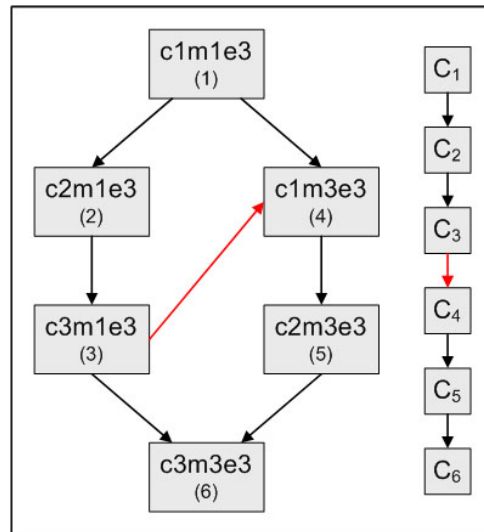


FIGURE 3.8 – Graphe de robustesse associé au cas de conception totalement ordonné représenté à la figure 3.4

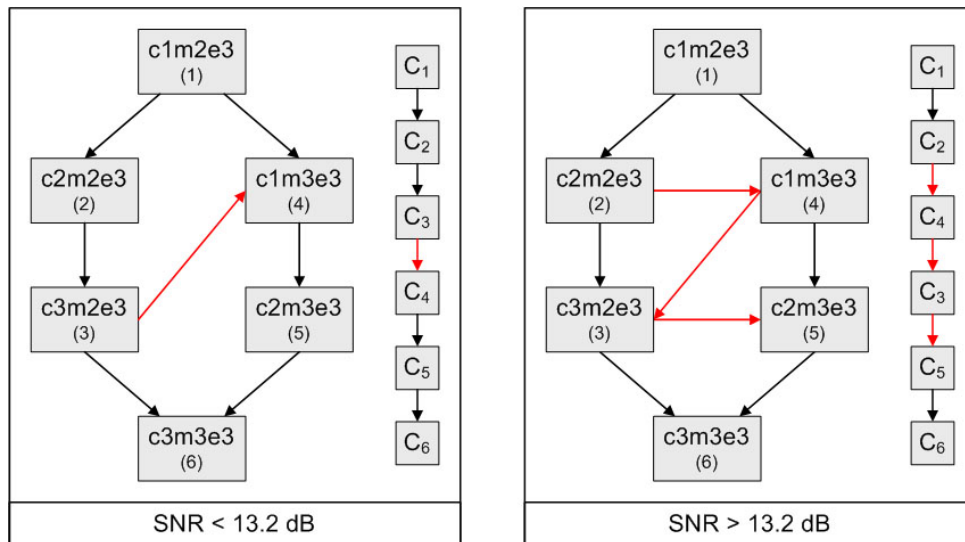


FIGURE 3.9 – Graphe de robustesse associé au cas de conception partiellement ordonné représenté à la figure 3.5

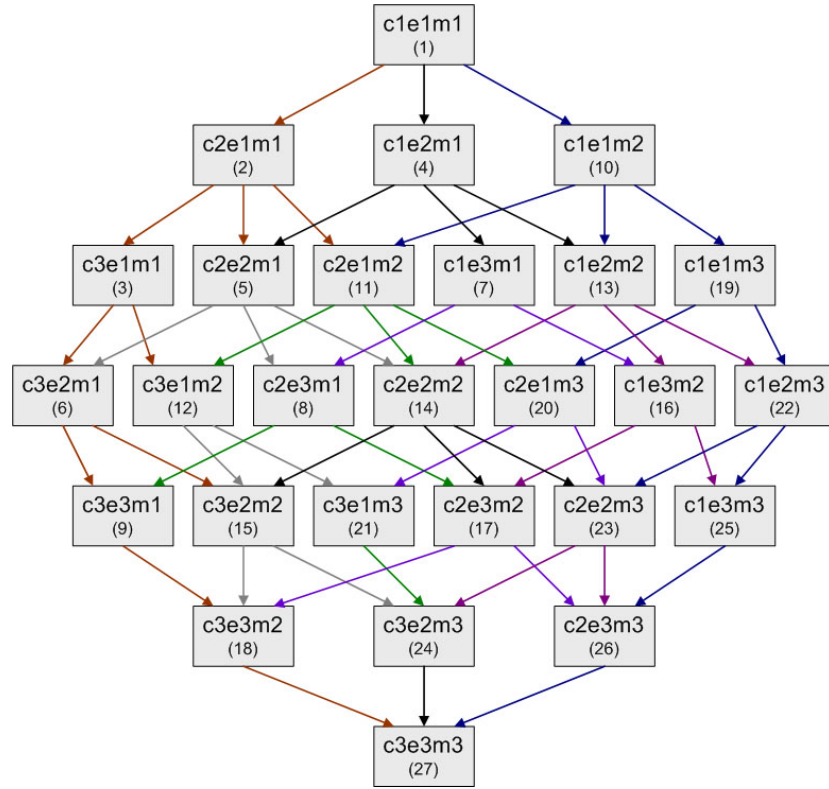


FIGURE 3.10 – Graphe de robustesse associé à un espace de conception impliquant toutes les configurations considérées dans le scénario de test

un large espace de conception. Il est donc intéressant de récolter un maximum d'informations pour faciliter la structuration de l'espace de conception. Comme nous l'avons suggéré dans la section 3.2.2, cette tâche peut être réalisée au préalable au travers de simulations hors ligne. Elle peut aussi être automatisée au niveau des stations de base qui récolteraient les résultats d'expérimentation de chaque radio pour construire un graphe plus complet permettant de diminuer encore plus le nombre de combinaisons. Les radios nouvellement connectées bénéficieraient ainsi d'une structuration plus forte de l'espace de conception donc elles pourraient explorer plus efficacement l'espace de conception à la recherche de la configuration optimale.

Le moteur cognitif s'appuie sur le graphe de robustesse pour construire son plan d'expérience en fonction des retours de l'environnement. Il évalue chaque configuration testée afin de vérifier si elle est compatible avec le service demandé. Si la contrainte est violée, le moteur cognitif parcourt le graphe afin d'identifier toutes les configurations moins robustes que la configuration testée. Ces configurations sont alors éliminées du plan d'expérience pour éviter de prendre une décision qui sera mauvaise à coup sûr. Si la contrainte est satisfaite, le moteur cognitif procède de la même manière sauf qu'il va parcourir le graphe à la recherche des configurations plus robustes que la configuration testée. Le moteur cognitif sait alors que ces configurations sont compatibles avec le service sans avoir à les tester. Il peut donc se focaliser sur d'autres configurations afin de découvrir des solutions potentiellement plus intéressantes.¹³

13. Imaginons par exemple que la configuration C_{14} vient d'être testée par le moteur cognitif. Si elle vérifie la contrainte, le moteur cognitif sait immédiatement que les configurations C_1 , C_2 , C_4 , C_{10} , C_{14} , C_5 , C_{11} et C_{13} feront partie elles aussi des configurations compatibles avec le service. Si C_{14} n'est pas suffisamment performante alors C_{15} , C_{17} , C_{23} , C_{18} , C_{24} , C_{26} et C_{27} ne le seront pas non plus.

De manière générale, le raisonnement et l'apprentissage seront plus faciles si la structuration de l'espace de conception est forte et adaptée au problème, et si les connaissances *a priori* sont importantes [111]. Il serait intéressant à ce titre de profiter des travaux de Weingart et al. présentés à la section 2.4.3. La méthode proposée permet en effet d'identifier les facteurs (ou combinaison de facteurs) ayant le plus d'influence sur les réponses observées. Le modèle prédictif dérivé est ensuite exploité par leur algorithme afin de guider l'exploration de l'espace de conception de manière structurée. Dans l'exemple donné dans [100], l'algorithme décide d'activer l'opération de codage/décodage canal lorsque le lien radio perd un trop grand nombre d'informations. Il sait en effet que ce bloc a le plus grand impact sur la fiabilité du lien radio compte tenu des paramètres modifiables disponibles. Notre moteur cognitif gagnerait à intégrer ces connaissances qualitatives dans le graphe de robustesse afin de pouvoir compter sur une structuration plus fiable de l'espace de conception.

3.4 Cadre d'apprentissage

Une radio peut bénéficier de capacités d'apprentissage pour améliorer ses performances en ligne. Cette faculté cognitive risque même de devenir nécessaire si l'on veut que les équipements continuent à évoluer de manière optimale dans ce paysage radio de plus en plus complexe.

L'apprentissage englobe toute méthode permettant de construire un modèle de la réalité à partir de données, soit en améliorant un modèle partiel (ou moins général), soit en créant complètement le modèle [111]. L'apprentissage peut se manifester sous la forme de plusieurs facultés énumérées dans [111]. Nous les reprenons ici dans le cadre d'une radio cognitive :

- *Entraînement* : Le moteur cognitif est entraîné pour rendre la communication toujours plus efficace. Il est capable d'analyser les expériences rencontrées pour préparer au mieux les activités de conception futures.
- *Reconnaissance* : Une radio cognitive est à même de reconnaître et de comprendre son contexte opérationnel afin de pouvoir répondre au mieux au problème de conception posé. Elle est capable notamment de reconnaître les autres utilisateurs dans son voisinage afin de savoir comment communiquer avec eux (identification de la fréquence centrale, de la bande passante, du débit symbole et du schéma de modulation) [106, 112, 90, 113, 114]. L'approche exploratoire adopte la méthodologie du raisonnement par cas afin de reconnaître des situations typiques et ainsi profiter des explorations passées pour identifier une meilleure solution plus rapidement. Notre approche vise quant à elle à reconnaître le problème de conception posé afin d'identifier toutes les configurations suffisamment performantes pour délivrer le service requis.
- *Généralisation* : Le moteur cognitif accumule de l'expérience en extrayant des règles de comportement à partir des exemples observés. Il peut alors bénéficier de cette abstraction des concepts pour appliquer ces règles à des situations non rencontrées précédemment. La généralisation sera alors réalisée avec plus ou moins de succès en fonction de la similarité du nouveau problème par rapport aux anciens.
- *Adaptation* : Une radio cognitive module ses interactions avec son contexte opérationnel afin de s'adapter aux différentes variations observées (*e.g.* cadre de régulation, environnement radio, application sélectionnée, durée de vie restante de la batterie). Elle s'assure de toujours choisir la configuration la plus en adéquation avec les informations récoltées.
- *Amélioration* : Une radio cognitive cherche continuellement à améliorer ses performances afin de délivrer le meilleur service possible à l'utilisateur tout en évitant une consommation excessive des ressources disponibles.

Les défis rencontrés par l'apprentissage artificiel sont très bien mis en valeur dans [83]. Notre moteur cognitif y répond déjà partiellement puisqu'il est capable d'apprendre en ligne et de façon autonome (*i.e.* sans aide extérieure). Les systèmes prédictifs utilisés sont évolutifs donc ils peuvent s'adapter continuellement à un environnement versatile. Le modèle prédictif dérivé est facilement interprétable donc il est facile pour un expert humain d'intégrer de nouvelles règles de conception (obtenues par exemple à partir de simulations hors lignes plus complètes). Le moteur cognitif s'appuie, de plus, sur des connaissances du domaine pour accélérer son apprentissage. Même s'il reste une grande place pour l'optimisation des mécanismes proposés¹⁴, notre solution a le mérite de s'attaquer aux problèmes critiques de l'apprentissage.

Dans un premier temps, nous présentons les différentes formes d'apprentissage et nous caractérisons le problème d'apprentissage posé par une radio cognitive. Nous évoquons ensuite les buts recherchés de l'apprentissage et leur mise en place dans notre solution.

3.4.1 Discussion sur les différentes formes d'apprentissage

L'apprentissage est réalisé par une procédure chargée d'estimer les paramètres associés au modèle prédictif, afin que celui-ci remplisse au mieux la tâche qui lui est affectée. Les situations d'apprentissage sont généralement classées en trois catégories selon la qualité des informations disponibles :

- *Apprentissage supervisé* : Lors d'un apprentissage supervisé, l'algorithme d'apprentissage dispose d'informations instructives sur les réponses attendues du modèle. Un "professeur" extérieur fournit des échantillons d'apprentissage sous la forme d'un couple entrée/sortie (d'où le terme supervisé). Les paramètres du modèle sont alors modifiés dans le but de minimiser le terme d'erreur entre la sortie obtenue et la sortie désirée. L'apprentissage correspond donc ici à un problème d'optimisation.
- *Apprentissage par renforcement* : Un agent intelligent ne peut pas toujours se baser sur des exemples pour guider son apprentissage. La seule information disponible se résume alors souvent à une simple indication (binaire ou scalaire) évaluant le comportement de l'agent dans son environnement. L'agent apprend ainsi à réaliser les actions appropriées en se basant sur cette évaluation issue de son interaction avec l'environnement. Cette remontée d'information sur la qualité des choix effectués se fait sans le recours d'un professeur capable de donner des instructions correctes (*i.e.* les actions à accomplir). L'agent se renforce donc en apprenant de ses échecs. Il se rapproche progressivement d'un comportement optimal en découvrant par lui-même les actions qui le mènent à la meilleure performance globale. Cette méthode d'apprentissage par essais/erreurs est une technique simple mais efficace, éprouvée chez l'homme et les animaux, permettant d'adapter le comportement d'un agent à son environnement. La tendance à sélectionner une action est renforcée si cette action est suivie d'un résultat satisfaisant ou si elle conduit à une amélioration (récompense). Dans le cas contraire, cette tendance est plutôt affaiblie ou inhibée (punition).
- *Apprentissage non supervisé* : Lorsqu'on ne dispose ni de professeur, ni d'instructions critiques, l'apprentissage ne peut se baser que sur l'ensemble des entrées supposées représentatives. On parle alors d'apprentissage non supervisé ou d'auto-organisation. Un système prédictif à apprentissage non supervisé évolue pour extraire les caractéristiques (ou régularités) des échantillons présentés sans avoir à connaître la sortie ou la classe associée. En d'autres termes, le système détecte et catégorise les caractéristiques persistantes des

14. Nous suggérons plusieurs améliorations et optimisations possibles au chapitre 6.

entrées sans aucun retour de l'environnement. Il développe sa propre représentation des stimuli d'entrée.

Le problème général d'apprentissage par renforcement est illustré à la figure 3.11, schéma (a) [115]. L'agent intelligent récupère des informations sur son environnement au travers de ses sens et il choisit une action pour influencer sur son environnement. Il reçoit un signal de renforcement de la part de l'environnement qui évalue son comportement sans lui indiquer le comportement correct. Le but de l'apprentissage est de maximiser le signal de renforcement cumulé dans le temps.

Le parallèle pour une radio cognitive est illustré au schéma (b) de la figure 3.11. Le moteur cognitif analyse les informations contextuelles récoltées par la radio afin de sélectionner la configuration la plus en adéquation avec le problème de conception posé. La configuration appliquée a, en retour, une influence sur le contexte opérationnel (*e.g.* le signal émis occupe une certaine portion du spectre, les algorithmes de traitement du signal consomme des ressources de calcul drainant l'énergie de la batterie). La radio récupère aussi des informations sur son comportement au travers des indicateurs de performance garants de la qualité du service délivré (*e.g.* BER).

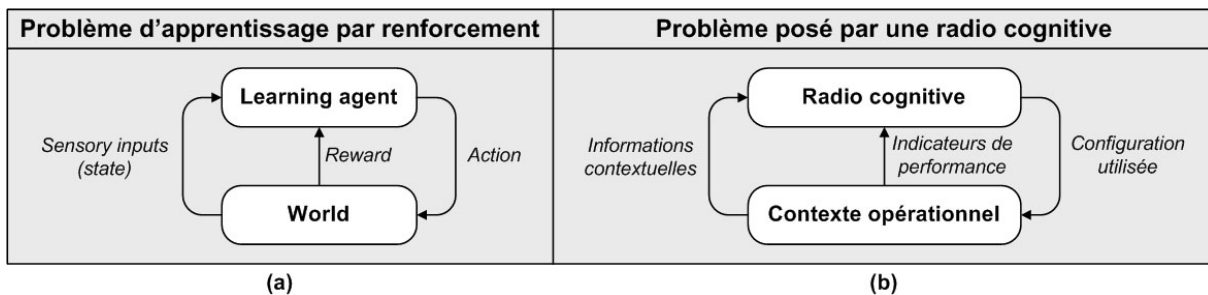


FIGURE 3.11 – Identification de la radio cognitive comme un problème d'apprentissage par renforcement

Le problème posé par une radio cognitive est typique d'un apprentissage par renforcement [116] comme le montre le parallèle précédent. Ce cadre d'apprentissage implique la problématique suivante.

L'agent intelligent cherche à identifier le mapping idéal entre la représentation d'une situation, symbolisée par un état de l'environnement, et son action optimale correspondante. Ce mapping correspond à la stratégie d'action de l'agent. La stratégie spécifie ce que l'agent doit faire dans chaque situation pour le niveau d'apprentissage courant. Elle peut être déterministe ou stochastique [117]. Une stratégie déterministe indique quelle action utiliser pour chaque état. Une stratégie stochastique associe à chaque état une distribution de probabilité sur un ensemble d'actions.

L'agent doit toujours maintenir un équilibre entre l'exploration (des régions encore vierges de l'espace de solution) et l'exploitation (des connaissances actuelles). Ses connaissances sont résumées dans la stratégie qu'il a développé jusque là. S'il décide de les exploiter, il choisit l'action jugée optimale par la stratégie dans son état d'apprentissage actuel. Mais l'exploitation à elle seule n'est pas nécessairement intéressante, surtout en début d'apprentissage où il est nécessaire de faire des essais pour avoir des indications sur les alternatives disponibles. Une exploration de l'espace des actions est donc aussi nécessaire. L'exploration peut se faire de deux manières : soit en choisissant une action qui n'a pas encore été essayée pour l'état visité, soit en sélectionnant une action jugée sous optimale par la stratégie actuelle. L'exploration introduit un risque sur

les performances de l'agent, l'action pouvant conduire à un faible signal de renforcement. Elle peut cependant permettre, dans des cas favorables, de trouver de meilleures actions et ainsi de se rapprocher progressivement de la stratégie optimale.

Nous avons porté une attention particulière sur la gestion du dilemme exploitation/exploration afin de limiter les risques d'expérimentation du moteur cognitif tout en lui donnant la possibilité de s'approcher de près du comportement optimal. Notre solution est décrite à la section 3.5.2 où nous présentons l'intérêt des automates d'apprentissage pour cette tâche.

3.4.2 Buts de l'apprentissage

Une radio cognitive augmente son autonomie décisionnelle en embarquant un processus d'apprentissage dans son moteur cognitif. Elle sait faire face à de nouvelles situations tout en étant capable d'améliorer ses performances sur des situations habituellement rencontrées.

De manière générale, le processus d'apprentissage peut être conçu suivant deux objectifs complémentaires [111] :

- *Modifier les connaissances de l'agent intelligent* : Cet objectif consiste à acquérir de nouvelles connaissances, à réviser des anciennes, voire même à en oublier certaines afin de permettre une prise de décision aussi fiable que possible sur l'ensemble des situations rencontrées. Cet objectif est poursuivi par les solutions associées à l'approche prédictive. Le moteur cognitif se base en effet sur une modélisation boîte noire pour apprendre les relations de causalité entre l'environnement radio et les performances des configurations. Cet objectif est aussi poursuivi dans une moindre mesure par l'approche experte (diffusion de nouvelles politiques d'action).
- *Rendre l'agent plus efficace relativement à la tâche qui lui est affectée* : Cet objectif peut être atteint par réorganisation, par optimisation ou par compilation [111]. Cet objectif est typiquement poursuivi par les solutions associées à l'approche exploratoire. Le moteur cognitif adopte la méthodologie du raisonnement par cas pour capitaliser sur ses expériences en compilant les tâches de conception les plus réussies dans l'espoir de répondre plus efficacement à des problèmes de conception similaires futurs.

Nous avons tenu compte de ces deux objectifs lors de la conception de notre moteur cognitif.

Les relations de causalité sont mémorisées à l'aide de systèmes connexionnistes évolutifs décrits de façon plus complète à la section 3.5.1. Le processus d'expérimentation s'appuie sur l'échelle de performance identifiée et sur les retours de l'environnement pour identifier les prédictions correctes, en phase avec les essais réalisés. Les systèmes prédictifs sont donc entraînés par apprentissage supervisé. La procédure d'apprentissage présente néanmoins des similitudes avec les algorithmes d'apprentissage non supervisé puisqu'elle fait intervenir une forme de compétition globale entre les neurones du système connexionniste (*cf.* sec. 4.2.1). Le système apprend à modifier sa structure afin de répondre aux échantillons d'entrée avec différentes parties du réseau. L'espace d'entrée est partitionné en plusieurs clusters. Chaque cluster regroupe un ensemble de points similaires associés à une classe préalablement identifiée lors d'une phase d'expérimentation passée (association déterministe). L'algorithme d'apprentissage permet de créer de nouveaux clusters (acquisition de connaissances) ou de modifier les paramètres de clusters existants (révision de connaissances). Nous avons, de plus, modifié sa version originale afin de donner la possibilité au système de désapprendre les associations sous-optimales (oubli de connaissances). Nous verrons à la section 3.5.1 que cette forme d'apprentissage hybride (supervisé/non supervisé) présente de nombreux intérêts pour une radio cognitive.

Notre moteur cognitif s'appuie sur une stratégie stochastique pour savoir quand déclencher

une phase d'expérimentation. L'expérimentation s'avère nécessaire en début d'apprentissage afin de permettre au moteur cognitif de caractériser les régions de l'espace de conception encore peu visitées. Avec l'expérience, le moteur cognitif "prend de l'assurance" dans ses prédictions. Il est donc préférable de recourir moins souvent à l'expérimentation afin de diminuer les risques de choisir ponctuellement une configuration trop peu performante. La stratégie développée assure une gestion efficace du dilemme exploitation/exploration. Elle est mise à jour en fonction des succès et des échecs de la phase d'expérimentation (apprentissage par renforcement). Le moteur cognitif encourage l'exploration si la phase d'expérimentation a permis de se rapprocher du comportement optimal. Il favorise l'exploitation si tous ses essais se sont montrés infructueux. Le compromis exploitation/exploration est contrôlé localement en associant un automate d'apprentissage à chaque cluster créé. Avec la maîtrise croissante du moteur cognitif sur les régions visitées, la probabilité locale d'exploration diminue progressivement au profit de la probabilité d'exploitation. Il faut faire attention cependant à ne pas réduire trop vite la probabilité d'exploration afin de donner une chance au moteur cognitif de se rapprocher localement du comportement optimal. Dans l'ensemble, le moteur cognitif devient de plus en plus efficace puisqu'il apprend à n'avoir recours à l'expérimentation seulement lorsqu'elle a une chance de succès. Cette stratégie lui permet ainsi de prendre une décision finale de plus en plus rapidement (en moyenne). La section 3.5.2 donne de plus amples détails sur la mise en place de cette stratégie et sa mise à jour.¹⁵

3.5 Techniques d'intelligence artificielle choisies

Introduit en 1994 par L. A. Zadeh, le "soft computing" trouve son inspiration dans l'esprit humain capable de raisonner et d'apprendre dans un environnement envahi d'incertitudes et d'imprécisions [118]. Ce nouveau paradigme contraste avec les techniques d'Intelligence Artificielle traditionnelles trop focalisées sur la précision, la certitude et la rigueur. Zadeh propose au contraire d'exploiter la tolérance pour les incertitudes, les imprécisions et les vérités partielles afin de proposer des solutions plus robustes, plus facilement implémentables, à faible coût (temporel, énergétique et financier) et en meilleur rapport avec la réalité.

Le soft computing est un partenariat de méthodologies distinctes conformes à son principe directeur. Ses principaux constituants sont la logique floue, les réseaux neuronaux, les méthodes d'optimisation (*e.g.* algorithmes génétiques, recuit simulé) et le raisonnement probabiliste, ce dernier incluant les systèmes chaotiques, les réseaux bayésiens et une partie de la théorie d'apprentissage. Dans ce partenariat, la logique floue permet de manipuler aisément des descriptions linguistiques à l'aide d'un raisonnement approximatif tenant compte des imprécisions inhérentes de l'environnement. Les réseaux neuronaux sont utilisés pour leurs capacités d'apprentissage et d'adaptation. Les méthodes d'optimisation se basent sur des métaheuristiques (souvent inspirées de la nature) permettant une recherche systématisée et aléatoire dans un espace complexe. Le raisonnement probabiliste assure la gestion des incertitudes et la propagation des croyances.

Les différentes méthodes peuvent être utilisées individuellement mais il est souvent intéressant de les combiner de façon synergique afin de pallier les inconvénients de l'une avec les avantages de l'autre. On parle alors de systèmes hybrides. Ces systèmes permettent d'intégrer plusieurs schémas de représentation des connaissances, plusieurs modèles de prise de décision et plusieurs

15. Nous avons aussi songé à un second mécanisme d'apprentissage pour rendre le moteur cognitif plus efficace dans ses prises de décision. Ce mécanisme se base sur la réorganisation et l'optimisation des bases de connaissances développées par les systèmes connexionnistes. Il sera présenté au chapitre 6.

stratégies d'apprentissage pour résoudre efficacement une tâche complexe [119]. Ils peuvent reproduire l'expertise humaine dans un domaine spécifique. Ils adoptent un mode de raisonnement approché afin de gérer les imprécisions et les incertitudes. Ils sont enfin capables de s'adapter et d'améliorer leurs performances dans un environnement bruité variant avec le temps.

Le tableau 3.4 permet de se rendre compte de l'intérêt d'une hybridation entre différentes techniques d'intelligence artificielle. Les systèmes neuro-flous [120, 121, 122, 123, 124, 125] sont les systèmes hybrides les plus populaires jusqu'à maintenant mais il existe bien d'autres exemples de synergie et les moyens de combiner les méthodes sont multiples [126, 127].

Critères	Systèmes experts	Systèmes flous	Réseaux de neurones	Algorithmes génétiques
Modélisation mathématique	–	+	--	--
Capacité d'apprentissage	--	--	++	–
Représentation des connaissances	++	++	--	--
Intégration de connaissances expertes	++	++	--	--
Acquisition automatique de connaissances	--	--	++	+
Non linéarité	--	+	++	++
Optimisation	--	--	+	++
Tolérance aux fautes	--	++	++	++
Tolérance aux incertitudes	--	++	++	+
Opération en temps réel	--	++	+	--
Interprétation des résultats	++	++	--	--

TABLE 3.4 – Étude comparative des atouts/faiblesses de plusieurs techniques d'intelligence artificielle (d'après [128])

Dans cette section, nous présentons les techniques d'intelligence artificielle que nous avons sélectionnées pour réaliser les tâches cognitives impliquées dans notre modèle de prise de décision autonome. Nous noterons d'ailleurs que ce modèle s'inspire très largement de la philosophie du soft computing de par son approche qualitative et approximative.

3.5.1 Prédiction des performances

Les **réseaux de neurones** sont un candidat de choix pour la prédiction des performances comme nous l'avons déjà justifié dans la section 2.4.3. Les structures connexionnistes traditionnelles souffrent néanmoins de certains inconvénients qui peuvent limiter leur intérêt dans un contexte de radio cognitive [129, 130, 131] :

- Le réseau est conçu avec une structure fixe qu'il est nécessaire de choisir *a priori* (*e.g.* nombre de neurones cachés dans la couche cachée d'un perceptron multi-couches).
- Un réseau mal conçu (*e.g.* avec un nombre trop grand de neurones cachés) est soumis au phénomène de surapprentissage. Les données d'apprentissage peuvent alors être approchées de trop près et le réseau perd en capacité de généralisation.
- Le réseau peut se retrouver piéger par un optimum local.
- Le réseau n'est pas capable de s'adapter à de nouvelles données sans oublier ce qu'il a appris à partir d'anciennes données (phénomène d'oubli catastrophique).

- Le réseau a besoin de plusieurs passages sur l'ensemble d'apprentissage pour correctement modéliser les données. La procédure d'apprentissage peut donc s'avérer longue et coûteuse en ressources de calcul.
- Le réseau distribue ses connaissances sur toute sa structure au niveau des poids des connexions. Il n'est donc pas facile d'interpréter comment le réseau a pu aboutir à une conclusion.

Ces inconvénients sont compensés en grande partie par le modèle que nous avons sélectionné : les **systèmes connexionnistes évolutifs**. Ce modèle s'inscrit dans un paradigme plus large aboutissant à la formation de systèmes intelligents capables de faire évoluer leur structure (expansion, réduction) en plus de l'adaptation traditionnelle des paramètres [132]. Cette attitude constructiviste augmente l'autonomie décisionnelle d'une radio cognitive en réduisant l'influence des choix effectués par le concepteur humain. La radio peut ainsi s'adapter à de nombreuses situations sans être mise à défaut par les capacités limitées d'une structure fixe.

Le modèle utilisé adopte une structure connexionniste reproduisant le fonctionnement d'un système expert flou [133] doté de capacités d'apprentissage incrémental. Le système est capable d'intégrer de nouvelles données en mettant à jour les règles existantes ou en modifiant sa structure pour mémoriser de nouvelles règles. Il sait aussi préserver ses capacités de généralisation en éliminant les règles erronées (node pruning) et redondantes (node aggregation). Chaque règle est interprétée comme un cluster associant une portion de l'espace d'entrée vers une fonction modèle appropriée [132]. Dans notre cas, cette fonction correspond à une classe donnée, identifiée lors de l'expérimentation en ligne (*cf.* sec. 3.2.3).

L'apprentissage est basé sur un ajustement local des éléments. Cette approche présente de nombreux avantages pour la prise de décision temps réel dans un environnement dynamique. Les modifications sont limitées seulement aux règles concernées par l'échantillon d'apprentissage traité donc l'adaptation en ligne est plus simple et plus rapide à calculer à chaque itération (surtout si la plateforme matérielle exploite le parallélisme inhérent de la structure). Les techniques de descente de gradient sont plus complexes en comparaison puisqu'elles modifient les connexions sur l'intégralité de la structure. De plus, un système évolutif offre un bon compromis entre stabilité (retenir les connaissances apprises à partir de données précédemment traitées) et plasticité (apprendre de nouvelles connaissances à partir de nouvelles données) ce qui lui permet d'éviter le phénomène d'oubli catastrophique couramment observé avec les structures connexionnistes optimisées par une procédure globale (*e.g.* rétropropagation du gradient).

Les systèmes évolutifs stockent les connaissances localement donc ils peuvent facilement accommoder leur structure afin d'intégrer une nouvelle classe (expansion de l'espace de sortie). Cette opération est beaucoup plus délicate à réaliser pour un réseau de neurones conventionnel distribuant ses connaissances sur l'intégralité de la structure. Cette caractéristique nous sera utile pour la prédiction de l'échelle de performance, le moteur cognitif découvrant de nouvelles relations d'ordre au fur et à mesure de ses expérimentations.

Le modèle sélectionné intègre des éléments de logique floue ce qui facilite la gestion des incertitudes liées à la phase d'observation de l'environnement. Les variables d'entrées sont quantifiées sur une échelle linguistique représentée par plusieurs ensembles flous. La figure 3.12 illustre par exemple notre représentation floue de la variable SNR à l'aide de deux fonctions de membre triangulaires traduisant le passage graduel entre de faibles valeurs de SNR vers des valeurs plus élevées. Les valeurs d'entrées sont qualifiées linguistiquement grâce à une fonction de fuzzification. La méthode de fuzzification la plus fréquente convertit la valeur d'entrée (non floue) en un singleton flou sur l'univers de la variable linguistique considérée. Un singleton flou est une valeur précise puisque l'ensemble flou est représenté par une fonction de membre égale à 1 pour la valeur

d'entrée et à 0 pour toutes les autres valeurs (figure a). Les incertitudes de mesure ne sont pas prises en compte dans ce cas. Cette méthode est néanmoins très populaire de part sa simplicité d'utilisation. Il peut être plus intéressant cependant d'utiliser une méthode de fuzzification plus complexe afin de tenir compte du bruit des mesures. La valeur d'entrée est alors convertie en un nombre flou [121], souvent à l'aide d'une fonction de membre triangulaire (figure b). Le sommet du triangle correspond alors à la valeur moyenne de la grandeur d'entrée et sa largeur est égale à deux fois la déviation standard de la grandeur. Cette opération permet de gérer les incertitudes directement dans le processus de prédiction sans recourir à une manipulation souvent complexe de variables aléatoires.

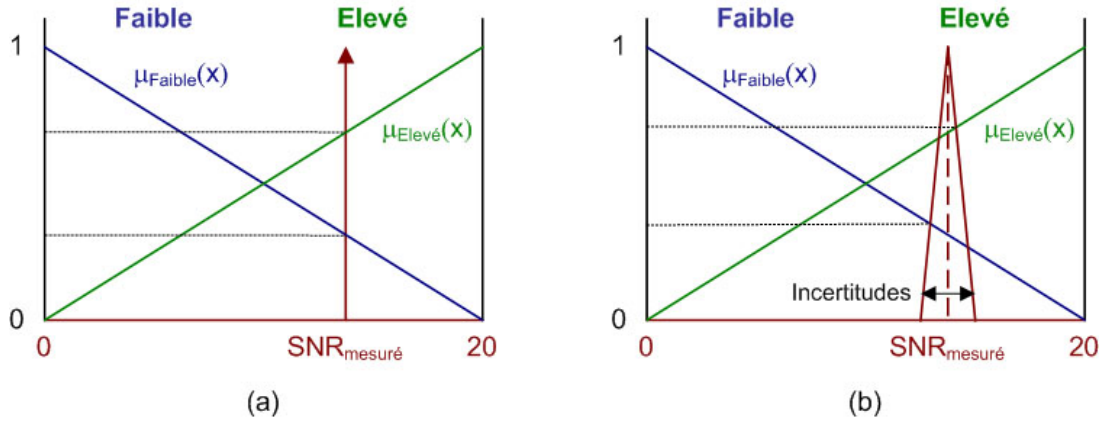


FIGURE 3.12 – Quantification linguistique associée à la variable SNR pour deux ensembles flous (SNR faible, SNR élevé) sans (a) et avec (b) prise en charge des incertitudes

L'introduction de l'échelle linguistique permet surtout une meilleure extraction des connaissances stockées dans la structure connexionniste. Chaque nœud de la couche évolutive peut ainsi être interprété sous la forme d'une règle floue compréhensible par un expert humain. Inversement, l'expert a aussi la possibilité d'insérer des connaissances préalables avant de déclencher le processus évolutif qui les raffinerait au fur et à mesure de son interaction avec l'environnement.

Les travaux de Kasabov *et al.* ont abouti à de nombreux modèles de systèmes connexionnistes évolutifs capables de traiter des problèmes de classification. Nous avons choisi le modèle ECF (Evolving Classification Function) pour sa simplicité.

Le modèle ECF est illustré à la figure 3.13. Il est composé de quatre couches. La première couche présente les variables d'entrée au reste de la structure. La seconde couche réalise une quantification floue des variables d'entrée. La troisième correspond à la couche évolutive mémorisant les règles identifiées lors des phases d'expérimentation passées. La dernière couche associe un nœud différent à chaque classe disponible.

Les algorithmes d'inférence et d'apprentissage seront décrits aux chapitres suivants pour la prédiction de la configuration compatible la moins robuste (sec. 4.2.1) et pour l'identification en ligne de l'échelle de performance (sec. 5.2.1).

3.5.2 Gestion du dilemme exploration/exploitation

L'exploration de l'espace de conception est nécessaire afin d'évaluer les configurations disponibles dans le but de se rapprocher du comportement optimal. L'exploration doit cependant stopper à un moment ou un autre et laisser place à l'exploitation pour éviter les erreurs résultant d'une expérimentation excessive.

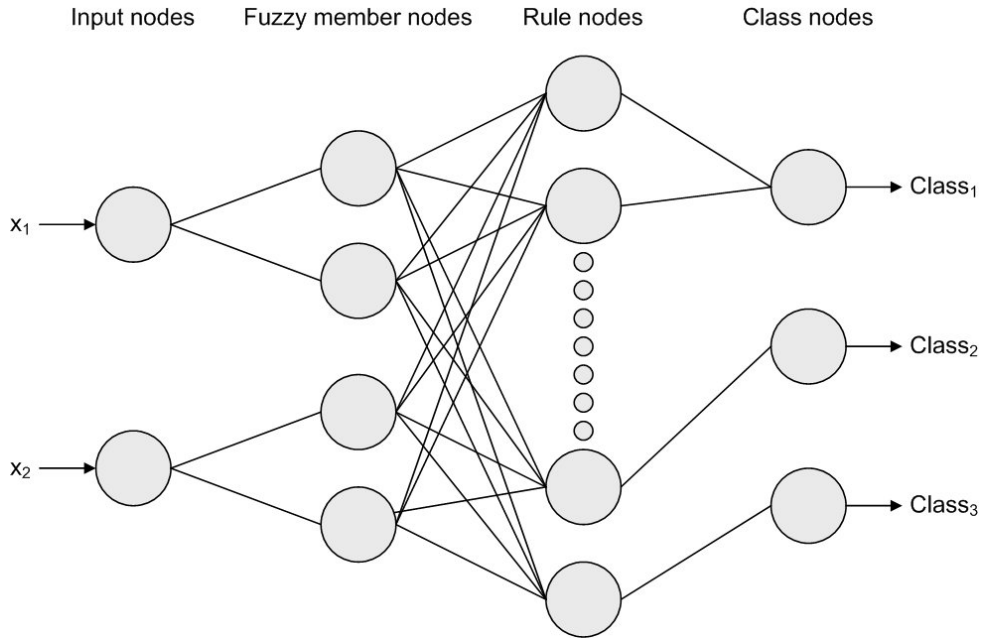


FIGURE 3.13 – Structure connexionniste du modèle ECF (Evolving Classification Function) [131]

Nous avons décidé de contrôler le dilemme exploitation/exploration localement à l'aide d'une équipe d'**automates d'apprentissage** s'appuyant sur la quantification de l'espace d'entrée réalisée par le système connexionniste. A chaque nœud de la couche évolutive, nous associons un automate d'apprentissage mémorisant la stratégie locale d'exploration. Le moteur cognitif privilégie l'exploitation avec une grande probabilité si le problème de conception est défini sur une région de l'espace bien caractérisée. L'exploration sera favorisée dans le cas contraire, le moteur cognitif déclenchant une phase d'expérimentation dans le but de découvrir de nouvelles configurations potentiellement plus satisfaisantes.

La stratégie de décision fait donc intervenir un modèle hybride basé sur les capacités d'auto-organisation des systèmes connexionnistes évolutifs et sur le mode de raisonnement des automates d'apprentissage. Une approche similaire a été proposée dans un contexte différent, plus théorique [134].

Un automate d'apprentissage est un outil de prise de décision adaptative apprenant à identifier l'action optimale à travers son interaction répétée avec l'environnement [135, 136, 137]. A chaque instant, il sélectionne une action sur la base d'une distribution de probabilité mise à jour par un algorithme d'apprentissage (ou schéma de renforcement) analysant les retours de l'environnement.

Un automate d'apprentissage est défini formellement par le quadruplet $\langle \alpha, \beta, p, T(\alpha, \beta, p) \rangle$ où α correspond à l'ensemble des actions disponibles, β représente l'ensemble des retours possibles par l'environnement et p mémorise les probabilités de réaliser chacune des actions compte tenu de l'état interne de l'automate. La fonction T est le schéma de renforcement modifiant le vecteur des probabilités en fonction de l'action réalisée et de la réponse reçue.

Dans notre moteur cognitif, chaque automate d'apprentissage n'a que deux actions disponibles (exploration ou exploitation) et l'algorithme de supervision retourne une évaluation binaire représentative du succès ou de l'échec de l'expérimentation tentée. L'expérimentation est un succès si elle permet de modifier les connaissances actuelles. Le moteur cognitif peut avoir progressé

dans l'échelle de performance et/ou il peut mettre à jour les relations d'ordre locales entre les configurations disponibles. L'expérimentation est un échec si le moteur cognitif n'a pris que des mauvaises décisions et s'il est conforté dans les relations d'ordre prédites.

Nous utilisons le schéma linéaire de renforcement caractérisé par l'équation 3.1 en cas de récompense et par l'équation 3.2 en cas de punition [135, 134] :

$$\begin{aligned} p_i(n+1) &= p_i(n) + a(1 - p_i(n)) \\ p_j(n+1) &= (1 - a)p_j(n) \quad j \neq i \end{aligned} \quad (3.1)$$

$$\begin{aligned} p_i(n+1) &= (1 - b)p_i(n) \\ p_j(n+1) &= \frac{b}{N-1} + (1 - b)p_j(n) \quad j \neq i \end{aligned} \quad (3.2)$$

Dans ces deux équations, a et b sont les paramètres de récompense et de punition respectivement et N correspond au nombre d'actions disponibles ($N = 2$ dans notre cas). Pour $a = b$, l'algorithme est appelé L_{R-P} (Linear Reward – Penalty), pour $b \ll a$, il est appelé $L_{R\epsilon P}$ (Linear Reward – ϵ -Penalty) et pour $b = 0$, il est appelé L_{R-I} (Linear Reward – Inaction). Ces algorithmes ont l'avantage d'être simples et ils garantissent une convergence vers des états satisfaisants (L_{R-P}) voire optimaux ($L_{R\epsilon P}$, L_{R-I}) [138]

Nous avons opté pour l'algorithme L_{R-P} pour ne pas multiplier le nombre de paramètres associés aux systèmes connexionnistes. Nous noterons, de plus, qu'une valeur trop faible de b multiplie les risques d'expérimentation (la probabilité d'exploration sera diminuée lentement en cas d'échec) même si elle assure la convergence vers une stratégie optimale. Or l'expérimentation peut avoir un impact négatif sur l'expérience de l'utilisateur donc il est préférable de minimiser les risques si elle ne peut apporter qu'un gain minime.

Les automates d'apprentissage travaillent dans l'espace des probabilités et non dans l'espace des actions. Cette particularité lui assure une meilleure tolérance au bruit par rapport à des modèles déterministes [137]. Notre gestion du dilemme exploration/exploitation implique deux sources de bruit. La première s'explique par l'impossibilité de tester toutes les alternatives lors d'une phase d'expérimentation. Le moteur cognitif doit savoir en effet arrêter son expérience de conception lorsqu'elle commence à dégrader la QoS sur une trop longue période. L'échec d'une expérimentation ne veut donc pas forcément dire qu'il n'est plus nécessaire d'explorer l'espace de conception. Cette remarque est d'autant plus vraie si l'espace de conception est large. La deuxième source de bruit est liée à la quantification de l'espace d'entrée. Le système connexionniste peut regrouper à tort certains problèmes de conception en estimant qu'ils sont similaires. L'expérimentation peut venir confirmer le modèle prédictif actuel pour un problème donné mais elle aurait pu donner des résultats différents pour un autre problème associé au même cluster. Ces deux considérations justifient notre intérêt pour la mise en place d'une stratégie stochastique pour la gestion du dilemme exploration/exploitation.

Pour minimiser l'influence de ces deux sources de bruit, nous avons introduit une notion de voisinage pour calculer la probabilité d'exploration associée au problème de conception considéré. Le moteur cognitif moyenne les probabilités associées aux clusters voisins du problème avant de décider s'il déclenche une phase d'expérimentation ou non (*cf.* sec. 4.2.5). Toute expérimentation réussie augmente ainsi les chances d'exploration pour le voisinage du problème traité. Ce résultat nous semble logique puisqu'une expérience concluante signifie que la région visitée n'était pas suffisamment bien caractérisée. Il y a donc des chances que son voisinage ne le soit pas aussi.

Le moteur cognitif profite ainsi de l'auto-organisation des systèmes connexionnistes pour construire une carte associant une région donnée de l'espace avec sa probabilité d'exploration.

La probabilité d'exploration est grande initialement puis elle diminue progressivement avec l'expérience du moteur cognitif.

3.5.3 Sélection de la configuration la plus adaptée

Le moteur cognitif dispose de toutes les informations pour prendre sa décision une fois l'ensemble des configurations compatibles prédites. Il s'appuie alors sur le formalisme de la **prise de décision multi-critères** pour sélectionner la configuration la plus adaptée. La qualité de la décision dépend bien évidemment de la fiabilité du modèle prédictif relativement au problème de conception traité.

Un problème de décision multi critères peut être exprimé de façon concise à l'aide d'un format matriciel représenté au tableau 3.5. La matrice de décision contient autant de lignes que d'alternatives $\{A_i\}_{i=1,\dots,m}$ et autant de colonnes que de critères $\{Cr_j\}_{j=1,\dots,n}$. La grandeur x_{ij} mesure la performance de l'alternative A_i vis-à-vis du critère Cr_j , cette mesure étant donnée sur une échelle cardinale ou ordinale. Chaque ligne représente les performances d'une alternative relativement à l'ensemble des critères considérés. Chaque colonne montre le contraste entre les alternatives disponibles vis-à-vis d'un seul critère. Des poids w_j sont associés à chaque critère pour traduire leur importance relative. On s'arrange généralement pour que leur somme soit égale à 1.

		Critères et poids associés				
		Cr₁	Cr₂	Cr₃	...	Cr_n
		w_1	w_2	w_3	...	w_n
Alternatives	A₁	x_{11}	x_{12}	x_{13}	...	x_{1n}
	A₂	x_{21}	x_{22}	x_{23}	...	x_{2n}
	\vdots	\vdots	\vdots	\vdots	\ddots	\vdots
	A_m	x_{m1}	x_{m2}	x_{m3}	...	x_{mn}

TABLE 3.5 – Matrice de décision multi-critères

La matrice de décision est mise au point par le système de notation s'appuyant sur le profil de l'équipement et les fonctions objectif développées pour récupérer/calculer les indicateurs de performance relatifs à chaque objectif secondaire. Le système de notation exploite ensuite la matrice formée afin d'agréger les performances et ainsi obtenir un score final symbolique de la satisfaction de l'alternative vis-à-vis de tous les critères. Les alternatives peuvent alors être comparées facilement afin d'identifier la configuration apparaissant comme optimale (alternative associée à la meilleure note).

De nombreuses méthodes existent mais aucune n'a été prouvée comme réellement supérieure aux autres [139, 140]. Le choix d'une méthode pour une autre dépend de la qualité des informations disponibles et des contraintes sur la complexité tolérable des traitements.

Nous avons décidé d'adopter la méthode TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) basée sur le concept suivant : l'alternative sélectionnée doit être la plus proche possible de la solution idéale et la plus éloignée possible de la pire solution. Pour appliquer cette méthode, il est nécessaire de définir une distance afin de comparer les solutions. La distance

euclidienne est très souvent utilisée. Nous décrivons l'algorithme complet à la section 4.2.2 en l'appliquant au scénario considéré.

Cette méthode a l'avantage d'être simple et facile à utiliser. La matrice de décision est normalisée donc il est possible de réaliser des compromis entre des indicateurs de performance de différente nature. De plus, la méthode s'adapte naturellement au contexte opérationnel puisqu'elle définit la solution idéale ainsi que la pire solution en fonction des alternatives rendues disponibles par l'environnement. Ces trois avantages sont particulièrement intéressants dans un contexte de radio cognitive.

Appliquée à notre scénario, la méthode donne des résultats cohérents avec l'intuition d'un concepteur humain donc nous n'avons pas cherché à complexifier davantage le système de notation embarqué. Des efforts de conception supplémentaires pourront néanmoins s'avérer nécessaires pour des scénarios plus complexes afin de s'assurer que le système de notation reflète au mieux la situation considérée. Le système de notation pourrait, par exemple, tenir compte des différences énergétiques entre une station de base et un terminal pour moduler les notes en fonction de la direction de la communication (montante ou descendante). Dans le cas d'une liaison montante, il pourrait favoriser des configurations à haut débit sans se soucier de la complexité des récepteurs associés. Pour des scénarios de conception multi-dimensionnels, le concepteur pourra profiter des travaux issus de l'approche exploratoire (notamment [19]) pour intégrer de nouvelles fonctions d'évaluation tenant compte, par exemple, des objectifs recensés à la colonne de droite du tableau 3.2. Les mesures de performance x_{ij} de certains objectifs peuvent aussi être mesurées en ligne au lieu d'être calculées à l'aide de fonctions paramétrées. La plateforme peut ainsi intégrer des sondes de consommation de puissance en entrée et en sortie de chaque bloc de traitement du signal pour représenter plus finement l'impact de la complexité algorithmique sur la consommation d'énergie. Enfin, les mécanismes de prise de décision devront certainement être adaptés pour tenir compte des imprécisions et des incertitudes liés à la phase de modélisation du contexte tout en restant économes en ressources de calcul. Nous pensons notamment que la logique floue présente un intérêt indéniable pour cette tâche [141, 140].

Mécanismes cognitifs pour la prise de décision autonome dans un espace de conception totalement ordonné

Sommaire

4.1	Organisation structurelle du moteur cognitif	66
4.2	Description des mécanismes cognitifs	68
4.2.1	Processus d'analyse prédictive	68
4.2.2	Processus de décision	71
4.2.3	Processus d'expérimentation	73
4.2.4	Processus d'apprentissage	73
4.2.5	Algorithme de supervision	76
4.3	Evaluation du moteur cognitif	77
4.3.1	Présentation des indicateurs de performance	77
4.3.2	Présentation des radios de référence	81
4.3.3	Résultats	82

Le moteur cognitif que nous proposons raisonne de manière autonome en s'appuyant sur les relations d'ordre définies par les échelles de performance et d'optimalité déduites dynamiquement en fonction du contexte. Il navigue le long de ces deux échelles pour :

- prendre une décision appropriée,
- s'adapter aux variations de son environnement,
- conduire une expérimentation maximisant les chances de progression vers un comportement optimal tout en limitant les risques d'erreurs,
- analyser les résultats d'expérimentation afin d'extraire de nouvelles règles de conception.

Les connaissances *a priori* sur la robustesse relative des configurations lui permettent d'explorer l'espace de conception plus efficacement. Le moteur cognitif exploite le graphe de robustesse embarqué pour inférer sur les tentatives et ainsi construire un plan d'expérience judicieux, cohérent avec les observations récoltées.

Les performances des configurations sont prédites approximativement par des systèmes évolutifs chargés d'estimer l'adéquation des configurations pour le service et l'environnement radio considérés. Les systèmes prédictifs s'auto-organisent en faisant évoluer leur base de connaissance afin de classer chaque problème de conception vers un ensemble de configurations suffisamment

performantes pour le résoudre. Le moteur cognitif profite ainsi des capacités d'apprentissage des modèles connexionnistes pour bâtir sa propre expertise de conception en fonction des expériences rencontrées. La nature approchée des prédictions assure une certaine tolérance aux erreurs de classification qui s'avère favorable pour la prise de décision dans un environnement complexe.

Le moteur cognitif développe en parallèle une stratégie stochastique d'exploration basée sur une équipe d'automates d'apprentissage. Cette stratégie lui permet de distinguer les situations en fonction de l'expérience qu'il a acquise au travers d'expérimentation passées. Notre moteur cognitif apprend ainsi à estimer la fiabilité de son modèle prédictif local afin de limiter les expérimentations aux régions de l'espace encore mal caractérisées.

Le chapitre précédent a retracé les réflexions ayant guidées la conception de notre moteur cognitif ainsi que les choix méthodologiques réalisés. Dans ce chapitre, nous détaillons les mécanismes cognitifs que nous proposons pour la prise de décision autonome d'une radio dans un espace de conception totalement ordonné. Dans un premier temps, nous présentons l'organisation structurelle du moteur cognitif puis nous décrivons les différents algorithmes employés et/ou développés. Les mécanismes cognitifs sont ensuite évalués à l'aide d'une étude de cas permettant de valider l'approche proposée.

4.1 Organisation structurelle du moteur cognitif

L'organisation structurelle du moteur cognitif est illustrée à la figure 4.1. Le schéma fait intervenir les trois éléments de la figure 2.2 (radio logicielle, moteur d'inférence, moteur cognitif) en précisant leurs interactions selon notre modèle de radio cognitive. La radio logicielle fournit des informations contextuelles au moteur cognitif qui les examine à l'aide du module CSM (*Context Sensing Manager*) s'appuyant en grande partie sur le moteur d'inférence. La phase d'analyse du contexte a été introduite en 3.1 lors de la présentation du scénario adopté.¹⁶

Le moteur cognitif a une organisation modulaire contrôlée par un algorithme de supervision que nous avons baptisé RALFE pour *Reason And Learn From Experience*. Cet acronyme fait référence à l'approche expérimentale de notre moteur cognitif faisant des tentatives pour se rapprocher du comportement optimal en apprenant de ses erreurs.

L'algorithme RALFE gère de nombreuses fonctions. Il assure l'activation et la coordination des modules. Il prend la décision d'expérimenter ou non en fonction de la probabilité locale d'exploration calculée par le module CAP (*Configuration Adequacy Predictor*). Il conduit l'éventuelle phase d'expérimentation à partir des retours du module RBA (*Radio Behavior Analyzer*) analysant le comportement de la radio. Il se charge aussi de sélectionner la configuration qui lui semble la plus adaptée en se basant sur :

- l'échelle d'optimalité dérivée par le module GS (*Grading System*),
- l'échelle de performance connue à l'aide de connaissances expertes (l'espace de conception est supposé totalement ordonné),
- la prédiction de la configuration compatible la moins robuste par le module CAP.

Enfin, il synthétise les résultats des expérimentations tentées afin de guider l'apprentissage du module CAP (révision/acquisition de règles de conception, mise à jour de la carte stochastique d'exploration).

16. Cette phase n'a pas fait l'objet d'une étude approfondie lors de cette thèse. Bien que le module CSM n'ait pas été implémenté, nous l'avons indiqué dans la figure pour des raisons de complétude mais aussi pour afficher les paramètres d'entrée de nos mécanismes cognitifs.

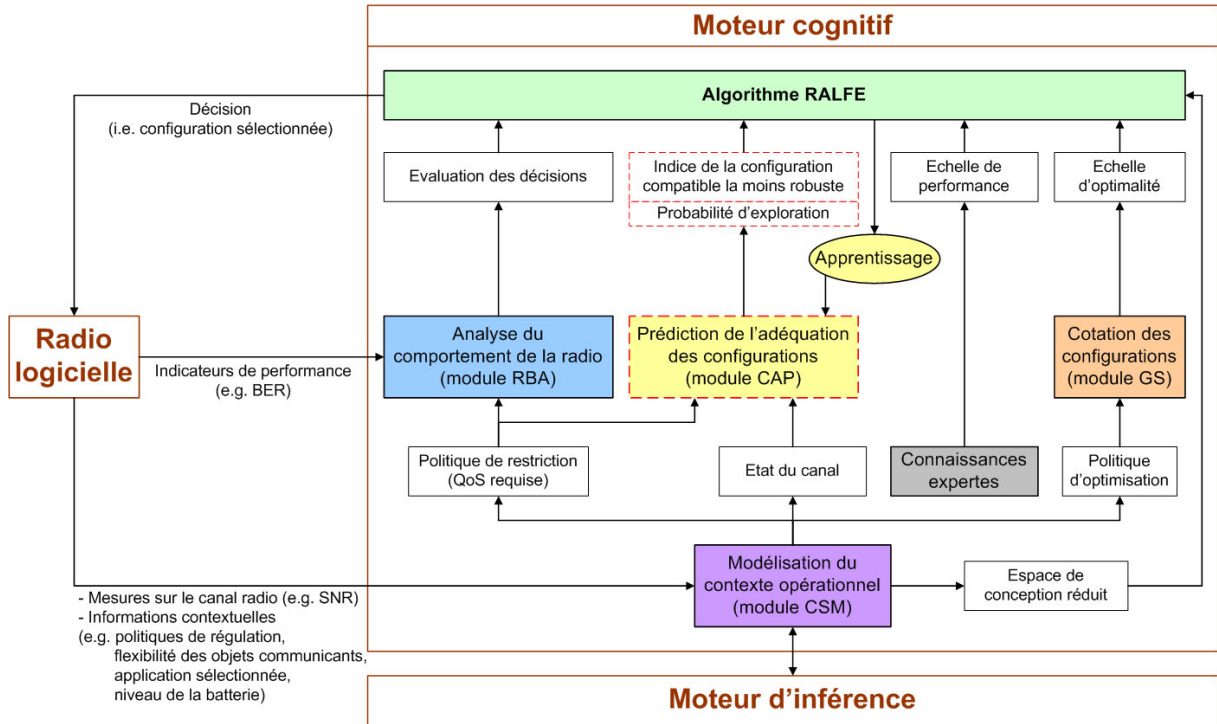


FIGURE 4.1 – Organisation structurelle du moteur cognitif

Le module GS tient compte de la politique d'optimisation pour attribuer une note à chaque configuration alternative en fonction de la préférence relative des objectifs secondaires. Le module CAP exploite les capacités d'inférence d'un système connexionniste évolutif pour réaliser sa prédiction à partir d'un modèle du canal (*e.g.* valeur du SNR) et de la spécification de la contrainte de QoS (le BER cible dans notre cas). Le module RBA connaît la valeur de cette contrainte pour pouvoir évaluer les décisions prises. Il renvoie un renforcement positif dès que la contrainte est vérifiée alors qu'il annonce une mauvaise décision si la configuration s'avère trop peu performante pour délivrer le service correctement. L'algorithme RALFE peut aussi tenir compte d'un espace de conception réduit en écartant simplement les configurations interdites (régulation, contraintes matérielles) lorsqu'il navigue le long des échelles identifiées.

Les mécanismes cognitifs exploitent les connaissances (embarquées et acquises) pour traiter les informations et les transformer en informations plus significatives pour la prise de décision. Les informations sont variables par nature puisqu'elles dépendent du contexte opérationnel (*e.g.* état du canal, politiques de restriction, HCR). Les connaissances embarquées correspondent aux fonctions objectif paramétrées et au graphe de robustesse. Le moteur cognitif acquiert aussi de nouvelles connaissances à travers son interaction avec l'environnement (règles de conception, stratégie d'exploration).

Dans la figure 4.1, le module CAP fait intervenir des capacités d'apprentissage donc ses connaissances évoluent avec l'expérience du moteur cognitif (module encadré en pointillés). Les autres modules s'appuient au contraire sur des connaissances invariables dans le temps (modules encadrés en trait plein). Le module CSM est censé réaliser une analyse experte des informations récoltées par la radio afin d'en déduire le problème de conception posé par le contexte opérationnel. Le module GS attribue des notes identiques lorsqu'il évalue le même ensemble de configurations pour une politique d'optimisation donnée. Le module RBA retourne la même

évaluation pour un même couple contrainte (*e.g.* BER cible) / performance obtenue (*e.g.* BER mesuré). Enfin, l'algorithme RALFE applique toujours le même raisonnement pour aboutir à une décision finale appropriée.

De la même manière, les informations fiables pour la prise de décision sont encadrées en trait plein. La fiabilité des informations encadrées en pointillé dépend quant à elle de la qualité du modèle prédictif développé par le moteur cognitif.

La fonction d'apprentissage est auréolée par un trait plein car les prédictions idéales sont connues avec certitude dès que le moteur cognitif déclenche une phase d'expérimentation. La certitude vient de la démarche exploratoire adoptée couplée à la structuration forte de l'espace de conception comme nous le verrons à la section 4.2.3.

4.2 Description des mécanismes cognitifs

L'organisation structurelle du moteur cognitif reflète l'implémentation C++ du moteur cognitif dans notre simulateur. Nous nous intéressons dans cette section à son organisation fonctionnelle pour mieux refléter les idées présentées tout au long du document. Nous rentrons notamment dans les détails algorithmiques de notre approche laissée à un niveau conceptuel à la section 2.5. Dans cette section, nous avons mis en valeur quatre processus caractéristiques de notre approche. Le processus de surveillance est laissé de côté pour le moment étant donné que nous travaillons avec un environnement de simulation statique (les problèmes de conception sont statiques lors de leur résolution par le moteur cognitif). Le processus de décision a été décomposé en deux afin de séparer l'analyse prédictive de la procédure de décision en elle-même. Les processus d'expérimentation et d'apprentissage sont maintenus tels quels.

Les quatre processus (analyse prédictive, décision, expérimentation, apprentissage) sont détaillés individuellement dans un premier temps. Nous présentons ensuite l'algorithme de supervision en charge de leur coordination.

4.2.1 Processus d'analyse prédictive

Le processus d'analyse prédictive a deux missions. Il se charge d'identifier l'ensemble des configurations compatibles avec le service compte tenu des conditions de propagation observées. Il cherche aussi à estimer le degré d'expérience du moteur cognitif par rapport à la région de l'espace de conception visitée. Cette expérience se quantifie de manière stochastique à travers les probabilités d'exploitation/exploration stockées localement au niveau des automates d'apprentissage.

Identification de l'échelle de performance

Dans un premier temps, le processus se base sur ses connaissances expertes pour identifier l'échelle de performance s'appliquant à l'environnement radio modélisé. Cette opération est réalisée en parcourant le graphe de robustesse à l'aide d'un algorithme de tri topologique (algorithme 1 ci-dessous). Deux versions de l'algorithme sont proposées. La version aléatoire appliquée de nombreuses fois permet de déterminer l'ensemble des échelles de performance compatibles avec un graphe donné. La seconde version assure un parcours du graphe en largeur traduisant un comportement prudent¹⁷ de la part du moteur cognitif. Dans les deux cas, l'algorithme progresse d'un sommet du graphe à un autre en s'appelant récursivement jusqu'à avoir ordonné l'ensemble

17. Le graphe est parcouru couche par couche en commençant par les configurations supposées plus robustes.

des configurations disponibles. L'espace de conception est totalement ordonné donc il n'existe qu'une seule échelle de performance compatible avec le graphe pour chaque état du canal. Les deux versions de l'algorithme produisent toujours le même classement dans ce cas. Ce classement représente alors une échelle fiable structurant parfaitement l'espace de conception au voisinage du problème considéré.

Algorithme 1 Algorithme de parcours du graphe de robustesse

```

 $PS \leftarrow$  Échelle de performance stockant les configurations par robustesse décroissante
 $L \leftarrow$  Ensemble des configurations susceptibles d'être sélectionnées par l'algorithme de tri

Insérer  $C_1$  dans  $L$  ( $C_1$  correspond à la configuration la plus robuste de l'espace de conception)

while  $L$  est non vide do
  repeat
    (version aléatoire)  $\rightarrow$  Choisir aléatoirement une configuration  $C_i$  dans  $L$ 
    (version prudente)  $\rightarrow$  Choisir la configuration  $C_i$  par ordre d'insertion dans  $L$ 
  until  $C_i$  n'a aucun parent parmi les configurations stockées dans  $L$ 
  Insérer  $C_i$  dans  $PS$ 
  for  $j = 1$  à  $N_i$  (nombre de liens entre  $C_i$  et ses configurations enfants) do
    if La connexion entre  $C_i$  et  $C_j$  est validée par l'état du canal radio (cf. sec. 3.3) et  $C_j$  n'a
    jamais été insérée dans  $L$  then
      Insérer  $C_j$  dans  $L$ 
    end if
  end for
  Éliminer  $C_i$  de  $L$ 
end while
  
```

Le processus fait ensuite fonctionner le système d'inférence associé au système connexionniste évolutif afin de :

- prédire l'indice de la configuration compatible la moins robuste,
- calculer la probabilité d'exploration pour le problème de conception traité.

Cette étape est entièrement réalisée par le module CAP. Son espace d'entrée est caractérisé par des statistiques représentatives du comportement du canal (*e.g.* SNR, dispersion temporelle) et par la contrainte de QoS considérée (*e.g.* BER cible).

Prédiction de la configuration compatible la moins robuste

Un nouveau problème DP_i est représenté par les valeurs associées aux N variables d'entrée $\{x_i^1, \dots, x_i^N\}$ du module. La première couche du système connexionniste normalise ces entrées sur le domaine de conception considéré (*e.g.* entre 0 et 20 dB pour le SNR, entre 10^{-2} et 10^{-6} pour le BER cible). La seconde couche les quantifie ensuite sur leur échelle linguistique respective. Les incertitudes ne sont pas prises en charge car nous supposons que le moteur cognitif bénéficie d'une estimation parfaite du canal. Les entrées sont donc confrontées directement à leurs ensembles flous comme nous l'avons illustré au cas (a) de la figure 3.12. Les paramètres des fonctions de membre sont calculés pour obtenir une partition forte des variables d'entrée [142] caractérisée par une division graduelle et homogène de l'univers de la variable (*cf.* fig. 3.12). Le problème de conception est alors défini par $DP_i = \{x_i^{f_1}, \dots, x_i^{f_M}\}$ où $x_i^{f_j}$ représente les entrées après fuzzification et M correspond à la somme des granularités (*i.e.* le nombre d'ensembles flous) de

chacune des variables. La couche évolutive permet ensuite de discrétiser l'espace d'entrée en un certain nombre de clusters significatifs pour la prise de décision. Chaque cluster Cl_k correspond à une règle définie par son centre $CC_k = \{c_k^{f_1}, \dots, c_k^{f_M}\}$ et son rayon R_k (domaine d'influence de la règle autour de son centre).

Le système connexionniste calcule la similarité du nouveau problème avec chacune des règles existantes à l'aide de la distance euclidienne locale normalisée¹⁸, définie par l'équation 4.1 :

$$d(DP_i, Cl_k) = d_k = \frac{\sqrt{\sum_{j=1}^M (x_i^{f_j} - c_k^{f_j})^2}}{\sqrt{\sum_{j=1}^M (x_i^{f_j} + c_k^{f_j})^2}} \quad (4.1)$$

La prédiction de la configuration compatible la moins robuste se fait alors à l'aide des heuristiques de décision du modèle ECF [131, 143] :

- Si $E_{act} = \{Cl_k, d_k < R_k\} \neq \emptyset$, l'ECF identifie le nœud le plus proche et sélectionne la classe associée à ce nœud.
- Si $E_{act} = \emptyset$, l'ECF identifie les M -of- N nœuds les plus proches, il calcule la distance moyenne à chaque classe et il prédit la classe associée à la distance moyenne la plus faible.

Les règles contenues dans E_{act} sont dites activées par le problème de conception traité. Le paramètre M -of- N est choisi par le concepteur du système. En pratique, nous l'avons pris égal à 3 comme il est courant de le faire [131].

Nous avons ajouté une heuristique supplémentaire pour gérer l'initialisation du moteur d'inférence lorsque la base de connaissances ne contient aucune règle. L'ECF retourne dans ce cas l'indice de la configuration la plus robuste (principe de précaution).

Estimation de la probabilité locale d'exploration

Pour estimer la probabilité locale d'exploration, nous avons défini une notion de voisinage caractérisée par la valeur du paramètre R_V (rayon de voisinage). Toutes les règles de distance inférieure à R_V sont alors impliquées dans le calcul de la probabilité d'exploration. Le voisinage est élargi aux M -of- N plus proches règles si la base de connaissances n'est pas suffisamment dense autour du problème de conception traité. L'ensemble des règles voisines $V = \{Cl_k, d_k \leq R_V\}$ contient donc toujours au moins M -of- N règles.

La probabilité locale d'exploration $P_{exploration}$ est obtenue en faisant la moyenne pondérée des probabilités d'exploration associées aux règles formant le voisinage du nouveau problème de conception (équation 4.2). Le poids d'une règle est d'autant plus grand que le centre du cluster est proche du problème traité.

$$P_{exploration} = \frac{\sum_{k \in V} w_k P_k}{\sum_{k \in V} w_k} \text{ avec } w_k = \exp\left(-\frac{d_k}{R_V}\right) \quad (4.2)$$

Lorsque la base de connaissances ne contient aucune règle, la probabilité d'exploration est initialisée à 1 car le moteur cognitif a besoin d'expérimenter pour commencer à construire sa propre expertise.

Le module CAP retourne la prédiction du système connexionniste directement sauf s'il estime que le problème n'a pas de solution parmi les configurations disponibles (*i.e.* il prédit la classe

¹⁸. Cette distance est conseillée pour le modèle connexionniste choisi mais il est possible aussi d'utiliser d'autres distances [131].

C_0). Ce cas particulier est traité avec précaution car le moteur cognitif peut décider de stopper sa communication temporairement pour économiser l'énergie de sa batterie. Il est donc préférable de s'assurer de la fiabilité de la prédiction avant de pouvoir l'exploiter avec certitude. Si la probabilité locale d'exploration est supérieure à un seuil P_{CC} , le module CAP estime que le moteur cognitif n'a pas encore assez d'expérience sur le problème traité donc il retourne C_1 la configuration la plus robuste de l'espace de conception. Si $P_{exploration} < P_{CC}$, le module considère comme fiable la prédiction de son système connexionniste et il maintient la configuration C_0 prédite.

4.2.2 Processus de décision

Le scénario de test adopté fait intervenir deux objectifs d'optimisation secondaires, à savoir le débit de transmission et la complexité algorithmique des traitements appliqués. Le module GS est donc conçu pour noter les alternatives en fonction de leur satisfaction intrinsèque vis-à-vis de ces deux objectifs.

Pour une configuration donnée, le débit d'information utile D est calculé à l'aide de la formule 4.3 [4] :

$$D = r \log_2(M) \quad (4.3)$$

où r est le rendement de la liaison et M représente l'ordre de la modulation employée.

Dans notre scénario de test, le calcul de la complexité algorithmique CA s'applique aux opérations de codage/décodage CA_{CC} et d'égalisation CA_{Equ} . Dans la continuité de notre approche qualitative, nous donnons une estimation de la complexité et non une valeur exacte.

L'équipement considéré peut activer un code bloc BCH(n, k, t)¹⁹ décodé à l'aide de l'algorithme de Berlekamp Massey²⁰. La complexité algorithmique est bien évidemment nulle ($CA_{CC} = 0$) si la configuration n'emploie pas de code correcteur d'erreurs.

Une configuration sans égaliseur n'introduit pas de complexité supplémentaire au niveau du récepteur. Sa complexité est évaluée à $CA_{Equ} = 1$ pour tenir compte de la prise de décision sur le symbole transmis. La complexité des traitements augmente si le récepteur intègre un égaliseur linéaire²¹ ou un égaliseur MLSE²² pour réduire les interférences inter-symboles induites par le canal à évanouissements sélectifs.

La complexité algorithmique d'une configuration est alors calculée en sommant la complexité individuelle de chaque bloc $CA = CA_{CC} + CA_{Equ}$.

Le module GS évalue la satisfaction intrinsèque des configurations supposées compatibles avec le problème uniquement. Il s'appuie sur le profil de l'équipement pour récupérer/calculer la matrice de décision alimentant la technique de prise de décision multi-critères (cf. sec. 3.5.3). La

19. n est la longueur des mots de code comprenant k bits d'information et $(n - k)$ bits de contrôle, t est le pouvoir correcteur du code.

20. Un code BCH($n, k, 1$) est décodé en $CA_{CC} = n - k$ opérations (calcul du syndrome). Un code BCH($n, k, t > 1$) nécessite $CA_{CC} = 2t(n - k) + t^3$ opérations car l'algorithme doit calculer $2t$ syndromes de $n - k$ bits puis résoudre une série de systèmes de taille maximale t .

21. La complexité algorithmique augmente de $\alpha L + (1 - \alpha)\sqrt{L}$ opérations, L étant la longueur de l'égaliseur. Le premier terme est lié au calcul des coefficients de l'égaliseur alors que le second terme représente l'opération de filtrage. Le paramètre α traduit le poids relatif de ces deux opérations sur la complexité de l'égaliseur. Il est pris égal à 0,99 par exemple. Pour une efficacité optimale du récepteur, la longueur de l'égaliseur doit être adaptée en fonction de la réponse impulsionnelle du canal radio.

22. L'égaliseur MLSE se base sur l'algorithme de Viterbi pour prendre une décision sur les symboles reçus [144]. Sa complexité dépend à la fois de l'ordre de la modulation utilisée (nombre d'état du treillis) ainsi que de la dispersion maximale du canal τ_{max} donnée en nombre de symboles (profondeur du treillis). Elle est égale à $CA_{Equ} = M^{\tau_{max}}$.

méthode TOPSIS sélectionnée suit alors la procédure suivante pour ordonner les configurations le long de l'échelle d'optimalité [140, 139] :

- Normalisation de la matrice de décision.

La valeur normalisée r_{ij} correspond à l'utilité relative de l'alternative A_i vis-à-vis du critère C_j . Elle est calculée à l'aide de la formule 4.4 :

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}, \quad i = 1, \dots, m, \quad j = 1, \dots, n \quad (4.4)$$

- Pondération de la matrice de décision normalisée.

Cette étape tient compte de la politique d'optimisation associant un poids à chaque objectif secondaire pour traduire leur importance relative. Les poids sont supposés normalisés (*i.e.* $\sum_{j=1}^n w_j = 1$). L'utilité pondérée est alors calculée à l'aide de la formule 4.5 :

$$v_{ij} = w_j r_{ij}, \quad i = 1, \dots, m, \quad j = 1, \dots, n \quad (4.5)$$

- Détermination de la solution idéale A^* et la pire solution A^- .

La solution idéale est une solution souvent fictive caractérisée par la meilleure performance atteignable pour tous les critères :

$$\begin{aligned} A^* &= \{\text{Critères à maximiser} \rightarrow \max_j v_{ij}, \text{Critères à minimiser} \rightarrow \min_j v_{ij}, \quad j = 1, \dots, n\} \\ &= \{v_1^*, v_2^*, \dots, v_n^*\} \end{aligned} \quad (4.6)$$

Inversement, la pire solution est caractérisée par les pires performances observées pour chacun des critères :

$$\begin{aligned} A^- &= \{\text{Critères à maximiser} \rightarrow \min_j v_{ij}, \text{Critères à minimiser} \rightarrow \max_j v_{ij}, \quad j = 1, \dots, n\} \\ &= \{v_1^-, v_2^-, \dots, v_n^-\} \end{aligned} \quad (4.7)$$

- Calcul des distances de séparation de chacune des alternatives avec les alternatives extrêmes.

La séparation entre A_i et la solution idéale est donnée par :

$$S_i^* = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^*)^2}, \quad i = 1, \dots, m \quad (4.8)$$

La séparation entre A_i et la pire solution est donnée par :

$$S_i^- = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^-)^2}, \quad i = 1, \dots, m \quad (4.9)$$

- Calcul de la proximité relative de la solution idéale.

$$C_i^* = \frac{S_i^-}{S_i^* + S_i^-}, \quad 0 < C_i^* < 1, \quad i = 1, \dots, m \quad (4.10)$$

- Rangement des alternatives.

Les alternatives sont rangées par ordre décroissant de C_i^* afin d'obtenir l'échelle d'optimalité.

La prise de décision est simple une fois l'échelle d'optimalité dérivée. Le processus n'a plus qu'à sélectionner la configuration de plus haut rang parmi les configurations compatibles.

4.2.3 Processus d'expérimentation

Le processus d'expérimentation est géré par l'algorithme RALFE s'appuyant sur l'échelle de performance et sur les analyses du module RBA pour proposer un plan d'expérience lui permettant de découvrir la stratégie optimale locale. L'espace de conception est totalement ordonné donc le processus peut naviguer avec fiabilité le long de l'échelle de performance pour identifier la configuration compatible la moins robuste. Il régresse le long de l'échelle si la configuration n'atteint pas les performances requises (attitude défensive) et il progresse vers des rangs plus élevés s'il décide de tester ses connaissances actuelles pour se rapprocher du comportement optimal (attitude offensive).

L'attitude défensive varie en fonction du paramètre A_{def} indiquant le nombre de rangs considérés lors d'une régression le long de l'échelle de performance. Une large valeur de A_{def} traduit un comportement conservateur, le processus d'expérimentation préférant revenir rapidement à des configurations connues comme plus robustes. Une faible valeur de A_{def} implique un plus grand risque de mauvaises décisions. Le processus d'expérimentation a en contrepartie moins de chances de s'éloigner du comportement optimal et la phase d'exploration est, par conséquent, plus efficace en moyenne.

L'attitude offensive est fixée, le processus d'expérimentation progressant un rang après l'autre dans l'échelle de performance. Nous avons décidé d'imposer ce comportement prudent pour minimiser les risques de mauvaises décisions. Une configuration sous-optimale reste une configuration correcte du moment qu'elle est suffisamment performante pour ne pas affecter l'expérience de l'utilisateur. Il est donc préférable de progresser lentement le long de l'échelle de performance même si la phase exploratoire peut prendre plus de temps, le moteur cognitif s'avérant moins audacieux.

Le processus d'expérimentation est activé après avoir testé la configuration associée au rang de plus haute compatibilité. Il suit alors un schéma réactif guidé par les renforcements estimés par le module RBA (cf. figure 4.2). Le renforcement est positif pour l'état *SC* (contrainte satisfaite ou Satisfied Constraint) et négatif pour l'état *VC* (contrainte violée ou Violated Constraint). Les réactions engendrées par le processus sont mises en valeur en trait fort dans la figure. Les mises à jour sur la connaissance du rang de plus haute compatibilité sont précisées en pointillé. Le processus d'expérimentation s'arrête lorsque le rang de plus haute compatibilité a été identifié. Le HCR est égal à 0 lorsque le problème de conception n'a pas de solution parmi les configurations disponibles. Le processus teste toujours la configuration la plus robuste avant de valider cette conclusion. Pour des problèmes moins difficiles, le HCR est identifié dès lors que la configuration de rang supérieur ne vérifie plus les contraintes. Pour des problèmes particulièrement simples (e.g. SNR élevé, canal très peu sélectif, service peu exigeant), le processus arrête l'expérimentation s'il réalise qu'il arrive à maintenir la QoS avec la configuration la moins robuste de l'équipement (i.e. de rang égal au nombre de configurations N_C disponibles).

4.2.4 Processus d'apprentissage

Le processus d'apprentissage met à jour la base de connaissances entretenue par le système connexionniste évolutif ainsi que la stratégie stochastique d'exploration. Il s'appuie sur les informations retournées par le processus d'expérimentation précisant la valeur du rang de plus haute compatibilité identifié $HCR_{expected}$ ainsi que l'indice de la configuration correspondante $(HCR_index)_{expected}$. Ces valeurs sont alors comparées à leurs prédictions respectives, i.e. $HCR_{predicted}$ et $(HCR_index)_{predicted}$, afin de guider l'apprentissage du modèle ECF et des automates concernés par le problème.

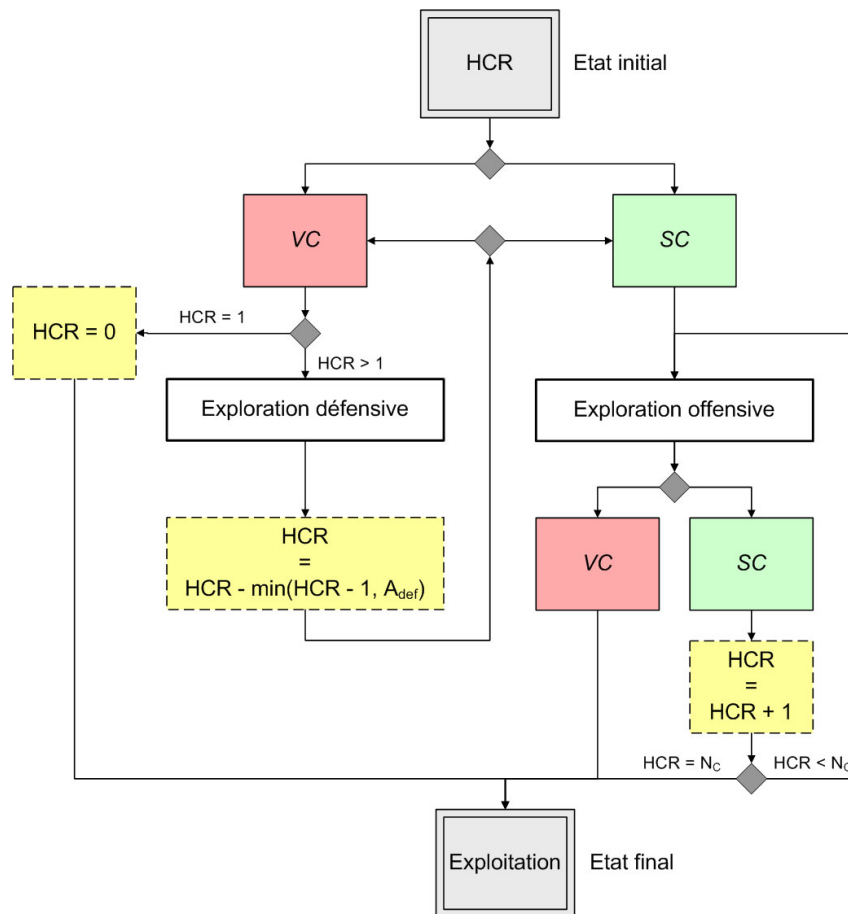


FIGURE 4.2 – Déroulement du processus d'expérimentation pour un espace de conception totalement ordonné

HCR : Highest Compatible Rank, VC : Violated Constraint, SC : Satisfied Constraint.

De manière générale, le processus d'apprentissage consolide les connaissances et il les généralise si les prédictions se sont avérées fiables. Il augmente aussi la confiance du moteur cognitif envers son modèle prédictif local (*i.e.* la probabilité d'exploration est diminuée au profit de la probabilité d'exploitation). Si le moteur cognitif a été induit en erreur par sa prédiction, le processus d'apprentissage révisé la base de connaissance pour intégrer les résultats déduits de l'expérimentation. La région visitée ne semble pas encore suffisamment maîtrisée donc le processus d'apprentissage préfère augmenter la probabilité locale d'exploration pour se donner une chance de mieux la caractériser.

La mise à jour de la stratégie locale d'exploration se base sur des informations obtenues à l'aide de l'algorithme 2 (encart ci-dessous). L'algorithme détermine quelle action renforcer entre l'exploitation et l'exploration. Il module aussi l'intensité du renforcement en fonction du succès de l'expérimentation. L'exploration est ainsi renforcée plus intensément si le processus d'expérimentation a progressé sur plusieurs rangs le long de l'échelle de performance.

Le processus d'apprentissage commence par adapter la probabilité d'exploration associée au problème de conception traité (eq. 4.2) en utilisant l'équation 3.1 en cas de récompense et l'équation 3.2 en cas de punition. La valeur corrigée de la probabilité sert de probabilité d'exploration initiale si le modèle ECF décide de créer une règle de conception supplémentaire pour représen-

Algorithme 2 Détermination des informations pour la mise à jour de la stratégie stochastique d'exploration dans un espace de conception totalement ordonné

```

if  $(HCR\_index)_{expected} = (HCR\_index)_{predicted}$  then
  L'exploitation est renforcée
  L'intensité de renforcement est égale à 1
else
  L'exploration est renforcée
  if  $HCR_{expected} > HCR_{predicted}$  then
    L'intensité de renforcement est égale à  $HCR_{expected} - HCR_{predicted}$ 
  else
    L'intensité de renforcement est égale à 1
  end if
end if

```

ter l'association nouvellement identifiée. Le paramètre de récompense/pénalité $a = b$ est obtenu en multipliant l'intensité de renforcement I_{reinf} par une valeur a_p choisie à l'initialisation du système :

$$a = b = a_p \times I_{reinf} \quad (4.11)$$

Une faible valeur de a_p traduit une volonté de pleinement explorer l'espace de conception afin de s'approcher au plus près du comportement optimal. La probabilité d'exploration diminue plus lentement donc le moteur cognitif fait plus souvent appel au processus d'expérimentation ce qui augmente les risques de mauvaises décisions. Une valeur trop grande de a_p peut laisser place en contrepartie à une exploitation excessive des connaissances conduisant à un comportement largement sous-optimal.

Les automates d'apprentissage associés aux règles voisines sont ensuite mis à jour afin de moduler la stabilité exploratoire de la région visitée. Les probabilités d'exploration sont ajustés de la même manière que précédemment sauf que le paramètre de récompense/pénalité est pondéré par les poids calculés lors du processus d'analyse prédictive (eq. 4.2) :

$$\forall Cl_k \in V, a_k = b_k = a_p \times I_{reinf} \times w_k \text{ avec } w_k = \exp\left(-\frac{d_k}{R_V}\right) \quad (4.12)$$

Les probabilités d'exploitation se déduisent des probabilités d'exploration par complémentarité à 1. Elles ne sont pas calculées en pratique afin de ne pas encombrer le moteur cognitif de grandeurs redondantes qui complexifieraient les processus sans aucun bénéfice.

Une fois la stratégie stochastique mise à jour, le processus d'apprentissage fait évoluer la base de connaissance du système connexionniste en appliquant un algorithme largement inspiré des heuristiques d'apprentissage du modèle ECF [131, 143, 145] :

1. Si la classe identifiée²³ n'est inférée par aucune règle de la base de connaissance actuelle (éventuellement vide), créer un nouveau nœud. La position du nouveau nœud (Cl_k) correspond aux coordonnées floues du problème de conception traité (DP_i) i.e. $c_k^{fj} = x_i^{fj}$ pour $j = 1, \dots, M$. Le rayon de son champ d'influence R_k est égal au paramètre R_{\min} représentant le rayon minimal d'un cluster (paramètre fixé à l'initialisation du système). L'automate d'apprentissage est initialisé avec la probabilité d'exploration calculée à partir du paramètre de renforcement de l'équation 4.11. L'algorithme s'arrête si un nœud est créé. Sinon, il passe à l'étape 2.

23. La classe identifiée fait référence ici à $(HCR_Idx)_{exp}$.

2. L'algorithme recense l'ensemble des règles activées par le problème $\{Cl_k \text{ t.q. } d_k < R_k\}$. Si cet ensemble est vide, il passe à l'étape 3. Il va à l'étape 4 dans le cas contraire.
3. L'algorithme élargit le champ d'influence des règles associées à la classe identifiée en prenant leur distance au problème comme nouveau rayon. Cette généralisation n'est autorisée que si le nouveau rayon reste inférieur au rayon maximal autorisé R_{\max} (paramètre fixé à l'initialisation du système) et si le nouveau champ d'influence ne couvre pas un nœud associé à une classe différente. Un nouveau nœud est créé comme à l'étape 1 si aucune des règles considérées ne peut vérifier les deux conditions ci-dessus.
4. Le champ d'influence des règles activées est réduit si elles sont associées à une classe différente de la classe identifiée. Le nouveau rayon est pris égal à la plus grande valeur entre $R - R_{\min}$ et R_{\min} . Un nouveau nœud est créé comme à l'étape 1 s'il n'existe aucune règle associée à la classe identifiée parmi les classes activées.

Nous avons observé de bons résultats malgré la simplicité des heuristiques employées par l'algorithme. De nombreuses améliorations sont cependant possibles comme nous le suggérons au chapitre 6. Nous avons aussi étudié l'influence des paramètres sur le comportement du moteur cognitif. Nous présentons nos conclusions sur ce point dans la section 4.3 consacrée à l'évaluation du moteur cognitif.

4.2.5 Algorithme de supervision

L'algorithme de supervision coordonne l'activation des différents processus afin d'aboutir rapidement à une décision de préférence optimale. Son fonctionnement est illustré à la figure 4.3.

Le moteur cognitif commence toujours par une analyse prédictive du problème de conception posé afin d'estimer l'impact de l'environnement radio sur la capacité des configurations à délivrer le service correctement. Le processus d'analyse prédictive calcule aussi la probabilité d'exploration locale afin de fournir une estimation grossière de la maîtrise du modèle prédictif sur la région de l'espace de conception visitée. L'algorithme de supervision se base alors sur cette probabilité pour décider aléatoirement s'il est nécessaire de mettre ses connaissances actuelles à l'épreuve (exploration) ou s'il est préférable de leur faire confiance (exploitation).

S'il favorise l'exploitation, le moteur cognitif lance le processus de décision afin d'identifier la configuration qui lui semble optimale. Le moteur cognitif sélectionne alors la configuration de plus haut rang dans l'échelle d'optimalité parmi les configurations supposées compatibles avec la contrainte de performance. Si cette configuration C_{opt} vérifie la contrainte, le moteur cognitif considère le problème comme résolu. Si la configuration n'est pas assez performante, il met à jour sa connaissance du rang de plus haute compatibilité en le faisant correspondre au rang de la configuration juste testée *i.e.* $HCR = PS(C_{opt}), PS(C_i)$ représentant le rang de la configuration C_i dans l'échelle de performance. Il active ensuite le processus d'expérimentation afin de réagir à sa mauvaise décision.

S'il privilégie l'exploration, le moteur cognitif teste la configuration compatible la moins robuste pour vérifier les connaissances établies. Il déclenche ensuite le processus d'expérimentation afin de réagir aux retours de l'environnement jusqu'à identifier la stratégie optimale locale. Il prolonge l'expérience de conception tant que le processus d'expérimentation l'estime préférable, voire nécessaire (*cf.* fig. 4.2). Il stoppe l'expérimentation dans le cas contraire puis il active le processus de décision en tenant compte des connaissances nouvellement acquises.

Le processus d'apprentissage n'est activé qu'à condition d'avoir obtenu de nouvelles informations à travers une phase d'expérimentation.

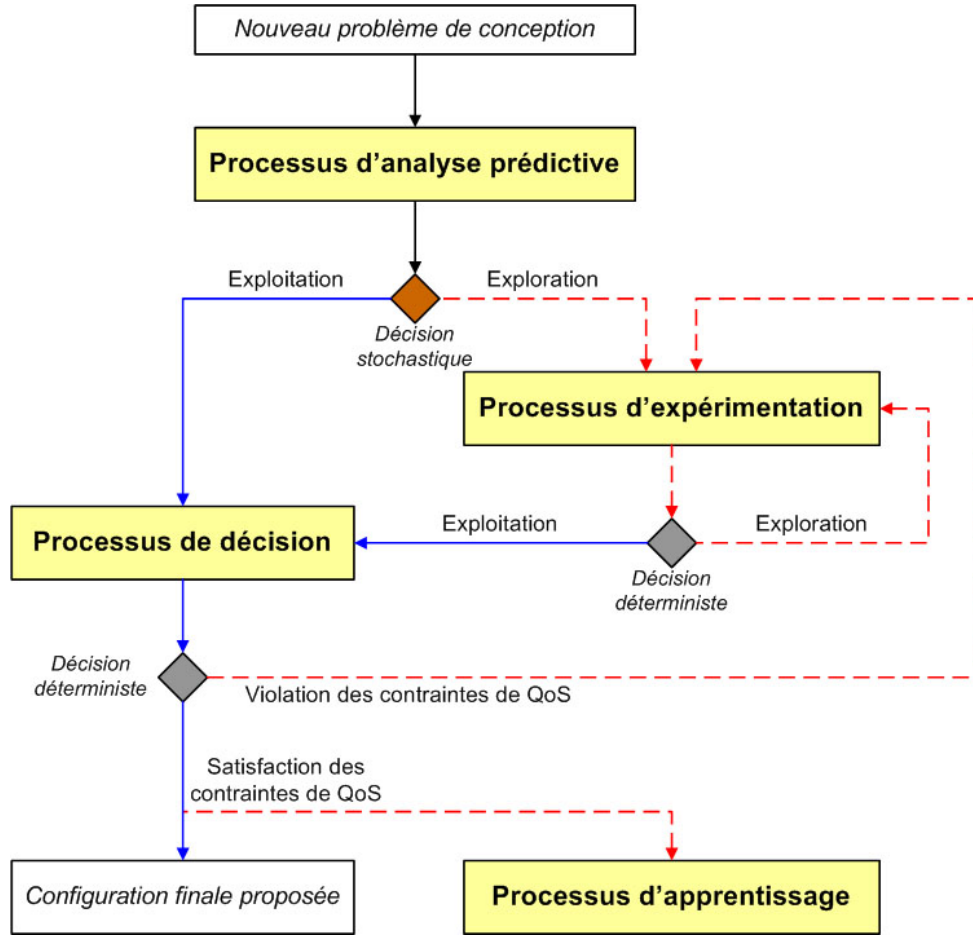


FIGURE 4.3 – Algorithme de supervision pour un espace de conception totalement ordonné
 Les liens en trait plein sont liés à l'exploitation des connaissances alors que les liens en pointillé sont liés à l'exploration de l'espace de conception.

4.3 Evaluation du moteur cognitif

Nous avons évalué les mécanismes cognitifs sur un cas d'adaptation du lien radio en présence d'un canal AWGN. La figure 4.4 précise le scénario de communication par rapport au schéma générique présenté à la section 3.1. L'égalisation n'est pas considérée ici puisque nous utilisons un modèle de canal idéalisé.

Nous avons défini un cadre d'évaluation pour caractériser complètement la solution que nous avons développée. Nous présentons dans un premier temps les différents indicateurs de performance identifiés. Nous introduisons ensuite des radios de référence afin de pouvoir comparer leurs performances avec celles de notre moteur cognitif. Nous démontrons enfin la pertinence de son comportement au travers des résultats obtenus pour un cas de conception totalement ordonné.

4.3.1 Présentation des indicateurs de performance

L'environnement de simulation mémorise de nombreuses informations sur le comportement du moteur cognitif au travers d'un observateur expert bénéficiant de connaissances parfaites sur l'ensemble des problèmes traités. L'expert profite des expressions analytiques pour calculer les performances exactes de chaque configuration en fonction de l'état du canal. Il connaît ainsi

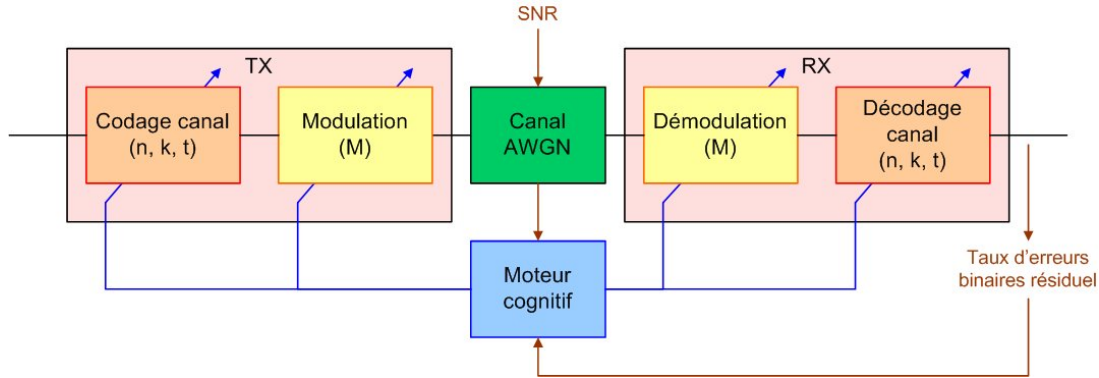


FIGURE 4.4 – Scénario de communication adopté pour l’étude de cas considérée

toutes les frontières de décision avec précision. Il peut donc analyser les différentes informations récoltées afin d’évaluer le moteur cognitif sur :

- son comportement général (*i.e.* l’efficacité de sa recherche de la configuration optimale),
- la fiabilité du modèle prédictif responsable de l’estimation des performances des configurations disponibles,
- la complexité de ce modèle,
- la pertinence de sa stratégie stochastique d’exploration,
- sa capacité à optimiser les indicateurs de performance de la radio.

Les différents indicateurs de performance identifiés sont présentés dans le tableau 4.1.

La matrice de confusion (CG_1) est un outil de visualisation de résultats couramment utilisé pour les problèmes de classification. Nous avons repris le concept en l’adaptant légèrement à notre modèle de prise de décision. La matrice de résultats est une matrice carrée contenant autant de lignes/colonnes que d’alternatives disponibles pour la prise de décision. Dans notre cas, le nombre d’alternatives est égal au nombre de configurations que l’équipement peut générer. Nous considérons aussi une dernière alternative pour traiter les cas d’incompatibilité entre le service demandé, les conditions de propagation observées et les configurations disponibles. Dans cette matrice carrée, l’indice de la ligne fait référence au rang associé à la décision optimale dans l’échelle de performance. L’indice de la colonne correspond au rang de l’échelle de performance associé à la décision proposée par le moteur cognitif en fin d’expérience de conception.

Le moteur cognitif peut avoir besoin de plusieurs décisions pour trouver une solution adaptée au problème. Il peut avoir à réagir à une erreur de prédiction et il peut aussi décider de mettre ses connaissances à l’épreuve dans une optique de perfectionnement. L’indicateur CG_2 comptabilise le nombre de problèmes résolus sans aucune mauvaise décision prise. Les expérimentations éventuelles du moteur cognitif n’ont alors aucun impact sur l’expérience de l’utilisateur. L’indicateur CG_3 évalue l’optimalité de la configuration adoptée comme solution finale au problème.

La distance au comportement optimal (CG_4) est mesurée en termes de rangs dans l’échelle d’optimalité. Elle est calculée en faisant la différence entre le rang de la décision optimale et le rang de la décision finale du moteur cognitif.

Les indicateurs CG_5 et CG_6 quantifient l’efficacité de la recherche d’une solution adaptée. L’indicateur CG_5 compte le nombre moyen de décisions prises par problème traité. L’indicateur CG_6 recense le nombre moyen de mauvaises décisions résultant d’une erreur de prédiction ou d’une tentative d’exploration ratée. Ces décisions dégradent la QoS temporairement donc il est préférable de limiter leur nombre au maximum.

Nom	Description des indicateurs de performance	AOS	AOT
Comportement général (CG)			
CG_1	Matrice de confusion	Y	N
CG_2	Pourcentage de problèmes résolus sans prendre de mauvaise décision	Y	Y
CG_3	Pourcentage de problèmes se terminant par la sélection de la configuration optimale	Y	Y
CG_4	Distance au comportement optimal	Y	Y
CG_5	Nombre moyen de décisions par expérience de conception	Y	Y
CG_6	Nombre moyen de mauvaises décisions par expérience de conception	Y	Y
CG_7	Pourcentage de problèmes traités sans activation du processus d'expérimentation	Y	Y
CG_8	Pourcentage de problèmes impliquant une phase d'exploration déclenchée pour s'assurer que le problème n'a pas de solution parmi les configurations disponibles	Y	Y
CG_9	Pourcentage de problèmes impliquant une phase d'exploration déclenchée à la suite d'une décision initiale erronée	Y	Y
CG_{10}	Pourcentage de problèmes impliquant une phase d'exploration déclenchée avec succès	Y	Y
CG_{11}	Pourcentage de problèmes impliquant une phase d'exploration déclenchée à tort	Y	Y
Fiabilité du modèle prédictif (FMP)			
FMP_1	Qualité moyenne des décisions prises par le moteur cognitif lorsqu'il exploite ses connaissances actuelles	Y	Y
FMP_2	Fiabilité moyenne des prédictions du module CAP	Y	Y
FMP_3	Carte d'adéquation associée au module CAP	$Y^{(1)}$	$Y^{(1)}$
Complexité du modèle prédictif (CMP)			
CMP_1	Nombre de règles contenues dans la base de connaissances développée par le module CAP	$Y^{(2)}$	$Y^{(2)}$
Pertinence de la stratégie stochastique d'exploration (PSSE)			
$PSSE_1$	Probabilité moyenne d'exploration sur l'ensemble de l'espace de conception	$Y^{(2)}$	$Y^{(2)}$
$PSSE_2$	Carte d'exploration	$Y^{(1)}$	$Y^{(1)}$
Bénéfices en termes de reconfiguration dynamique (BRD)			
BRD_1	Pourcentage de problèmes résolus à l'aide d'une configuration permettant de maintenir la QoS requise	Y	N
BRD_2	Débit moyen d'information utile sur l'ensemble de la simulation	Y	N
BRD_3	Complexité algorithmique moyenne sur l'ensemble de la simulation	Y	N

TABLE 4.1 – Indicateurs de performance pour l'évaluation du moteur cognitif

AOS : Calcul de l'indicateur de performance sur l'ensemble de la simulation (Averaged Over Simulation)

AOT : Calcul de l'évolution de l'indicateur de performance au cours de la simulation (Averaged Over Time)

Y/N : Yes/No

(1) Les paramètres de la couche évolutive sont analysés à travers une série de clichés pris à divers instants de la simulation. Le cliché obtenu en fin de simulation correspond à la valeur AOS de l'indicateur de performance.

(2) Ces indicateurs de performance traduisent déjà une progression donc il est inutile de les moyenner. La valeur AOS de CMP_1 correspond au nombre de règles dans la base de connaissances du module CAP en fin de simulation.Le même raisonnement s'applique pour $PSSE_1$.

Des indicateurs de performance supplémentaires (CG_7 à CG_{11}) ont été définis pour analyser la réponse du moteur cognitif aux problèmes de conception posés. Le moteur se contente d'exploiter ses connaissances pour un grand nombre de problèmes lorsque CG_7 est élevée. Ces décisions peuvent néanmoins s'avérer sous-optimales pour certains problèmes qu'il n'aurait pas assez explorés. L'indicateur CG_8 traduit la prudence du moteur cognitif préférant déclencher une phase d'exploration pour s'assurer que le problème n'a aucune solution parmi les configurations disponibles. Le moteur cognitif teste alors la configuration la plus robuste avant de conclure à une incompatibilité éventuelle entre les exigences de QoS, les conditions de propagation et les possibilités de reconfiguration de l'équipement. Le modèle prédictif contient encore des erreurs tant que CG_9 n'est pas nulle. Les erreurs restantes sont liées aux régions de l'espace de conception encore mal caractérisées. Le moteur cognitif généralise alors de façon trop brutale et choisit une configuration initiale trop peu performante. Il réagit à cette erreur par une phase d'exploration doublée d'une phase d'apprentissage afin de mémoriser l'association nouvellement identifiée. Les indicateurs CG_{10} et CG_{11} indiquent que le moteur cognitif a déclenché une exploration offensive en pensant que son modèle prédictif était sous-optimal. Cette décision est correcte en moyenne si CG_{10} prend une valeur plus élevée que CG_{11} . Elle introduit des erreurs inutiles dans le cas contraire (risques d'expérimentation).

L'indicateur FMP_1 est incrémenté si le moteur cognitif identifie la configuration optimale en exploitant les connaissances accumulées. L'indicateur FMP_2 est incrémenté si le module CAP prédit la valeur exact du HCR pour le problème traité. La comparaison entre ces deux indicateurs permet d'illustrer la tolérance aux erreurs de prédiction. Le moteur cognitif est capable de prendre une décision optimale même si le module CAP se trompe dans sa prédiction. Il suffit pour cela que la configuration correspondante fasse partie de l'ensemble des configurations prédites comme compatibles (*cf.* sec. 3.2).

La carte d'adéquation (FMP_3) représente l'espace de classification tel qu'il est décomposé par le système connexionniste du module CAP. Une classification idéale permettrait de retrouver les frontières de décision illustrées à la figure 3.4.

La complexité du processus d'analyse prédictive dépend directement de la taille de la base de connaissance développée par le module CAP. L'indicateur CMP_1 fournit donc une estimation grossière de la complexité de ce processus.

L'indicateur $PSSE_1$ montre l'évolution du moteur cognitif dans sa gestion du dilemme exploration/exploitation. Il est calculée en moyennant les probabilités locales d'exploration mémorisées par l'équipe d'automates d'apprentissage.

La carte d'exploration ($PSSE_2$) fournit la probabilité d'exploration locale pour chacune des régions de l'espace de conception. La probabilité devrait diminuer progressivement avec le temps au fur et à mesure que le moteur cognitif gagne de l'expérience sur son espace de conception. Elle risque cependant de diminuer moins vite au voisinage des frontières de décision, l'espace de conception étant plus difficile à caractériser pour ces régions.

Les indicateurs BRD_1 à BRD_3 sont obtenus en moyennant les indicateurs de performance de la radio pour les configurations proposées comme solution finale par notre moteur cognitif. Ces indicateurs ne tiennent donc pas compte de la phase de recherche d'une configuration adaptée. Nous discuterons néanmoins de l'impact de cette phase sur la QoS lors de la présentation des résultats.

Les indicateurs de performance sont évaluées sur une fenêtre intégrant L_W problèmes de conception consécutifs. Pour calculer les indicateurs sur l'ensemble de la simulation (Averaged Over Simulation ou *AOS*), il suffit de prendre $L_W = N_{DP}$, N_{DP} correspondant au nombre de problèmes de conception générés par l'environnement de simulation. Pour observer leur évolution

au cours de la simulation (Averaged Over Time ou *AOT*), nous considérons une fenêtre glissante de taille $L_W < N_{DP}$. Les indicateurs sont alors calculés sur les i problèmes de conception tant que $i \leq L_W$. La fenêtre intègre ensuite les L_W problèmes de conception DP_{i-L_W+1} à DP_i pour $i > L_W$.

Il n'est pas toujours possible ou intéressant d'observer l'évolution d'un indicateur. Nous avons précisé lorsque c'était le cas dans le tableau 4.1.

4.3.2 Présentation des radios de référence

Les performances de notre moteur cognitif sont comparées à une radio cognitive idéale connaissant l'impact exact du canal radio sur les performances des configurations disponibles. La radio cognitive idéale est capable de prendre une décision optimale sans jamais avoir recours à l'expérimentation pour identifier les règles de conception s'appliquant à son contexte opérationnel. Notre moteur cherchera progressivement à se rapprocher du comportement de cette radio.

Nous avons aussi introduit deux autres radios de référence volontairement plus élémentaires que notre modèle de radio cognitive. Ces deux radios expérimentent jusqu'à avoir identifié la configuration optimale avec certitude. Elles ne bénéficient par contre d'aucune mémoire donc leur comportement reste le même tout au long de la simulation. La comparaison des performances de ces radios avec la notre permet alors de valider la qualité de notre modèle prédictif enrichi progressivement grâce aux capacités d'apprentissage du système connexionniste évolutif.

La première radio procède par force brute en testant toutes les configurations avant de prendre une décision. La seconde radio exploite le même graphe de robustesse que notre moteur cognitif afin de mieux contrôler son exploration de l'espace de conception.²⁴ Cependant, elle applique toujours le même plan d'expérience obtenu à l'aide de la version prudente de l'algorithme de parcours du graphe (algorithme 1). Lors de son expérience de conception, la radio s'appuie une nouvelle fois sur le graphe de robustesse pour inférer sur les décisions prises. Elle arrête son exploration lorsqu'elle connaît l'impact du canal radio sur toutes les configurations disponibles (soit par test, soit par inférence). Sa décision est donc forcément optimale.

La comparaison des performances de ces deux radios de référence a pour but de montrer l'apport du graphe de robustesse embarqué pour l'expérimentation.

Les indicateurs de performance présentés au tableau 4.1 sont pour la plupart spécifiques à notre modèle de radio cognitive. Seuls les indicateurs CG_2 à CG_6 sont pertinents pour les radios de référence ainsi que BRD_1 à BRD_3 .

CG_2 est égale à 100% pour une radio cognitive idéale qui prendrait la décision optimale toujours du premier coup. Cet indicateur dépend du problème de conception pour les deux autres radios de référence. Si toutes les configurations vérifient les contraintes de performance, le problème de conception est résolu sans aucune mauvaise décision. Nous considérons aussi qu'un problème sans solution ne donne lieu à aucune mauvaise décision. Chaque configuration tentée est alors vue comme une décision correcte puisque le moteur cognitif n'a aucune chance de vérifier les exigences de QoS. La valeur de CG_2 est toujours la même pour ces deux radios de référence qui s'efforcent de caractériser entièrement la région de l'espace de conception visitée.

CG_3 est égale à 100% pour les trois radios de référence tout au long de la simulation puisque les radios connaissent les performances de chaque configuration ou elles le déterminent par une expérimentation poussée (plus ou moins intelligente). De la même manière, la valeur de CG_4 est toujours égale à 0 pour ces trois radios.

24. Cette radio de référence opère donc par force brute assistée.

CG_5 est égale à 1 pour la radio cognitive idéale. Elle doit être calculée séparément pour les deux autres radios de référence. Pour la radio opérant par force brute, le nombre de décisions prises est égal au nombre de configurations disponibles. Une décision supplémentaire est nécessaire si la dernière configuration tentée ne correspond pas à la configuration optimale. Pour la radio plus intelligente, il est nécessaire de simuler le plan d'expérience sur l'ensemble des problèmes pour pouvoir calculer la valeur de CG_5 .

Les bénéfices de la reconfiguration dynamique peuvent être évalués au travers des indicateurs de performance BRD_1 à BRD_3 . La comparaison avec les radios non flexibles permet d'évaluer si notre moteur cognitif arrive à maintenir la QoS exigée tout en optimisant les ressources radio et énergétiques. La comparaison avec la radio cognitive idéale permet d'estimer l'impact de la construction d'un modèle prédictif fiable sur les différents indicateurs de performance.

4.3.3 Résultats

Le moteur cognitif est testé sur le scénario de reconfiguration présenté à la figure 3.4. L'espace de conception est totalement ordonné grâce aux connaissances expertes encodées sous la forme du graphe de robustesse exposé à la figure 3.8. L'environnement de simulation génère aléatoirement 20000 problèmes de conception qu'il soumet au moteur cognitif afin d'observer son comportement avec l'expérience accumulée.

Les résultats présentés sont moyennés sur 10 simulations puis analysés en détail à l'aide du cadre d'évaluation mis en place. L'évolution des indicateurs de performance est calculée sur une fenêtre de longueur $L_W = 1000$. Le moteur cognitif ne contient aucune règle dans sa base de connaissances initiale et il est configuré avec les paramètres suivants :

- Comportement défensif (A_{def}) : Le moteur cognitif régresse rang par rang le long de l'échelle de performance en cas de violation de contrainte ($A_{def} = 1$).
- Seuil de vérification pour les problèmes supposés sans solution (P_{CC}) : Ce paramètre intervient uniquement lorsque le module CAP indique une incompatibilité entre les configurations disponibles, l'environnement radio et les exigences de QoS. Comme indiqué à la section 4.2.1, le moteur cognitif teste la configuration la plus robuste si la probabilité locale d'exploration est supérieure à 0,1 et il maintient sa prédiction dans le cas contraire ($P_{CC} = 0, 1$).
- Rayon d'influence maximal (R_{max}) : Le rayon d'influence d'un cluster ne peut dépasser $R_{max} = 0, 4$.
- Rayon d'influence minimal (R_{min}) : Le rayon d'influence d'un cluster ne peut descendre en dessous de $R_{min} = 0, 1$. Ce paramètre correspond aussi à la valeur initiale du rayon pour un nouveau cluster. Il contrôle enfin la réduction du champ d'influence des clusters en cas de prédiction erronée.
- Nombre de règles impliquées dans le processus d'inférence ($M-of-N$) : Si le problème n'a activé aucune règle, le système connexionniste utilise les trois nœuds les plus proches pour produire sa prédiction ($M-of-N = 3$).
- Rayon du voisinage (R_V) : Le processus d'analyse prédictive tient compte de tous les nœuds situés à une distance inférieure à 0,2 lorsqu'il calcule la probabilité d'exploration locale ($R_V = 0, 2$).
- Paramètre de récompense/punition (a_p) : Les automates d'apprentissage sont mis à jour en considérant $a_p = 0, 1$.

Qualité des décisions prises par notre moteur cognitif

Le tableau 4.2 représente la matrice de confusion obtenue sur l'ensemble de la simulation. Dans ce tableau, les décisions optimales sont mises en valeur en gras et les mauvaises décisions en italique. Les éléments restants sont associés à des décisions correctes. La matrice s'interprète ligne par ligne. Par exemple, la troisième ligne recense le nombre de problème de conception résolu de manière optimale par la configuration C_2 . Notre moteur cognitif a donc pris une décision optimale pour 1080,1 problèmes, une décision correcte pour 158,4 problèmes et aucune mauvaise décision.

4352.3	<i>0</i>	<i>0</i>	<i>0</i>	<i>0</i>	<i>0</i>	<i>0</i>
<i>105.1</i>	2404.6	<i>0</i>	<i>0</i>	<i>0</i>	<i>0</i>	<i>0</i>
<i>0</i>	158.4	1080.1	<i>0</i>	<i>0</i>	<i>0</i>	<i>0</i>
<i>0</i>	16.2	199.5	6031.9	<i>0</i>	<i>0</i>	<i>0</i>
<i>0</i>	0	0	58.7	851.6	<i>0</i>	<i>0</i>
<i>0</i>	0	0	58.8	59.7	750.7	<i>0</i>
<i>0</i>	0.1	0	95.9	9.1	154.8	3612.5

TABLE 4.2 – Matrice de confusion pour le cas étudié

Par construction, notre moteur cognitif continue à réagir tant qu'il n'a pas identifié une configuration compatible avec la contrainte (à moins que le problème n'ait pas de solution parmi les configurations disponibles). Il est donc impossible qu'une expérience de conception se termine par une configuration de rang supérieur au rang optimal (matrice triangulaire supérieure). Il n'existe en fait qu'une seule situation problématique. Elle apparaît lorsque le moteur cognitif estime à tort que le service ne peut être délivré correctement alors qu'il existe une configuration vérifiant la contrainte (première colonne). Ce cas arrive rarement en pratique (105,1 problèmes sur 20000 traités). Il peut être évité complètement en réduisant le seuil P_{CC} à des valeurs plus faibles. La configuration la plus robuste C_1 serait alors testée beaucoup plus régulièrement, surtout pour les problèmes proches de la frontière de décision. Pour des problèmes particulièrement difficiles, cette décision supplémentaire diminue la réactivité du moteur cognitif qui pourrait décider de changer de bande fréquentielle ou de réduire les contraintes de QoS plutôt que de tester C_1 inutilement.

La matrice de confusion montre que notre moteur cognitif apprend à prendre des décisions adaptées juste en interagissant avec son environnement (sa base de connaissances est vierge de toute règle initialement). A la fin des expériences de conception, les décisions sont optimales (diagonale de la matrice) ou correctes. De plus, les décisions correctes sont proches de l'optimum (cases juste à gauche de la diagonale). Cette première analyse a permis de valider le bon fonctionnement des mécanismes cognitifs.

Les figure 4.5 à 4.7 complètent cette analyse en montrant la progression du moteur cognitif tout au long de la simulation.

La figure 4.5 montre l'évolution du pourcentage de problèmes traités sans qu'aucune mauvaise décision ne soit prise (CG_2). Nous pouvons constater une légère discontinuité au niveau du 1000^{ème} problème généré. La progression plus rapide du moteur cognitif résulte en fait d'un artifice de calcul lié à la longueur de la fenêtre glissante. Les problèmes traités en début de simulation disparaissent progressivement de la fenêtre de calcul au-delà du 1000^{ème} problème. Au début de la simulation, le moteur cognitif dispose d'une expérience très limitée donc il a tendance à activer

le processus d'expérimentation pour accumuler des connaissances sur les règles de conception à appliquer. L'exploration s'accompagne très souvent d'une erreur de décision car le moteur cognitif continue de progresser le long de l'échelle de performance tant qu'il n'est pas mis en défaut par ses choix (*i.e.* la configuration sélectionnée atteint les exigences de QoS). Les problèmes sont donc résolus sans mauvaise décision si :

- le problème est suffisamment facile pour que la contrainte soit vérifiée par toutes les configurations disponibles,
- le problème n'a pas de solution parmi les configurations disponibles (les décisions prises sont alors considérées comme correctes puisqu'aucune configuration ne peut atteindre les exigences de QoS),
- le processus d'expérimentation n'est pas activé.

La figure 4.5 montre que notre moteur cognitif se comporte de mieux en mieux après avoir accumulé de l'expérience. Il profite des capacités d'apprentissage de son système connexionniste pour améliorer son expertise sur l'espace de conception. Il apprend aussi à limiter les risques d'expérimentation en actualisant sa stratégie d'exploration en fonction des renforcements reçus. En fin de simulation, 94,8% des problèmes sont traités sans affecter aucunement l'expérience de l'utilisateur. Les ratios de référence procédant par force brute restent quant à elles à une valeur légèrement au-dessus de 40%.

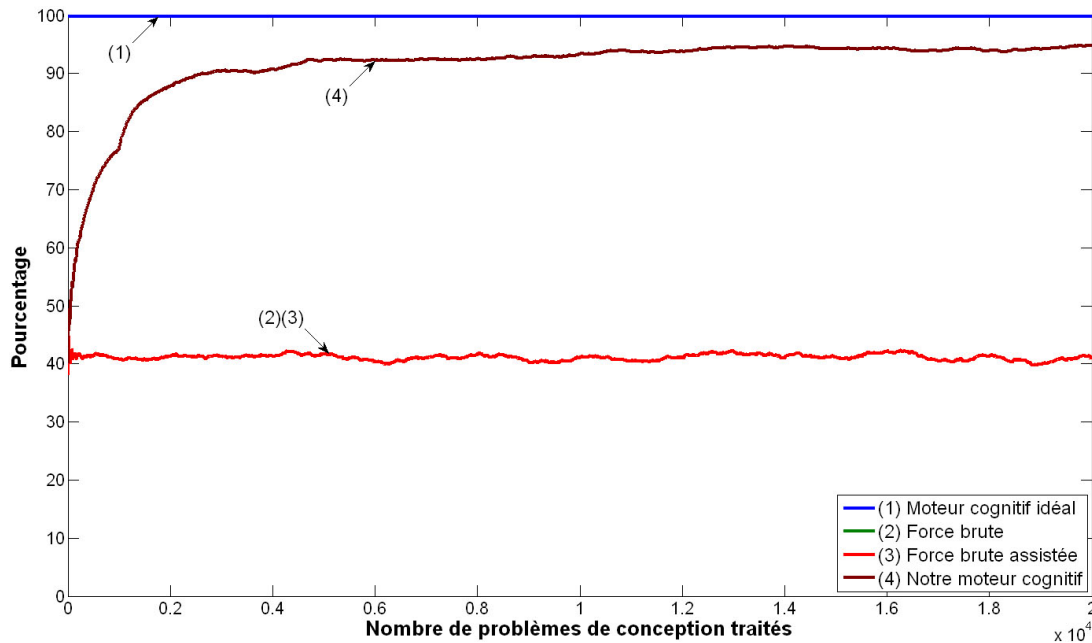


FIGURE 4.5 – Pourcentage d'expériences de conception n'impliquant aucune mauvaise décision lors de la phase de recherche d'une configuration adaptée (CG_2)

La figure 4.6 montre que notre moteur cognitif reste proche du comportement optimal tout au long de la simulation (CG_3). Sur l'ensemble de la simulation, 95,4% des expériences de conception proposées se terminent par une décision optimale. La figure 4.7 confirme cette tendance en affichant l'évolution de la distance moyenne au comportement optimal (CG_4). La valeur observée (légèrement supérieure à 0,05) indique que le moteur cognitif ne se trompe en moyenne que d'un seul rang dans l'échelle d'optimalité pour 20 problèmes traités. Ces résultats témoignent de la qualité des décisions prises par notre moteur cognitif en fin d'expérience de conception.

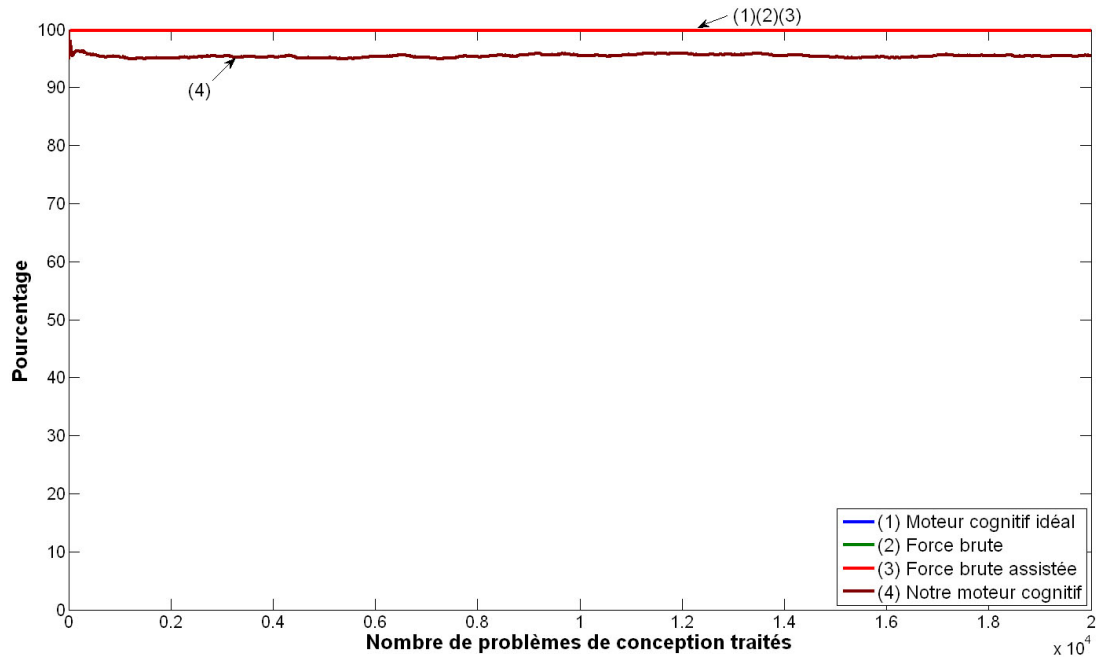


FIGURE 4.6 – Pourcentage d’expériences de conception se terminant par une décision optimale (CG_3)

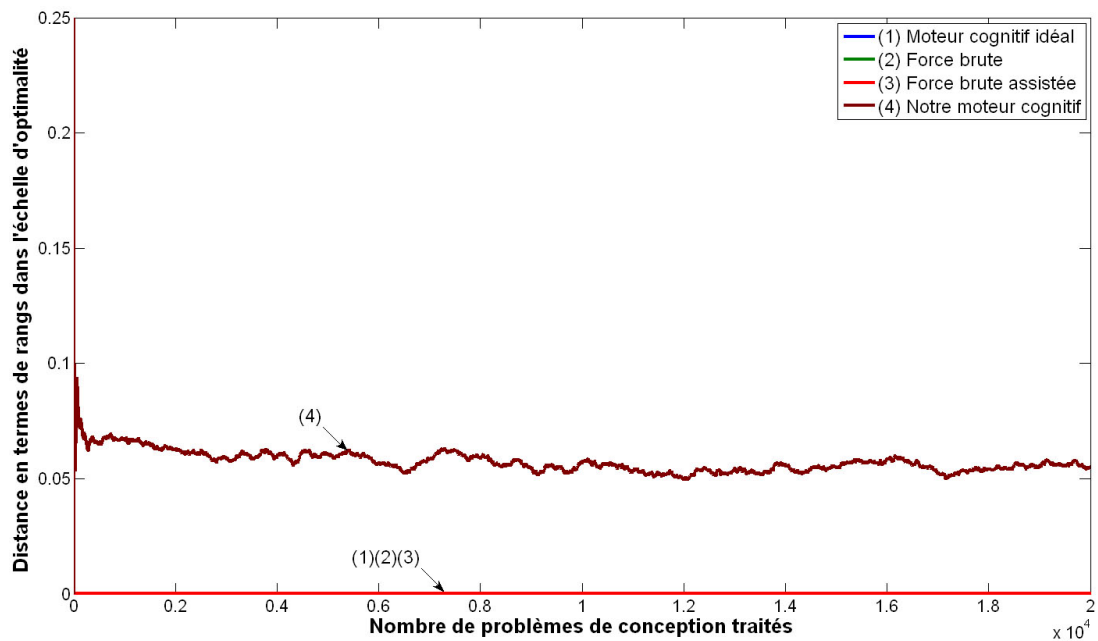


FIGURE 4.7 – Distance moyennée au comportement optimal en termes de rangs dans l’échelle d’optimalité (CG_4)

Efficacité du moteur cognitif dans sa recherche d'une configuration adaptée

La figure 4.8 affiche l'évolution du nombre moyen de décisions prises par expérience de conception (CG_5). Le moteur cognitif ne dispose pas d'un modèle prédictif fiable en début de simulation. Le processus d'expérimentation est donc souvent activé pour réagir aux erreurs de prédictions et rechercher les configurations optimales. Avec l'expérience, les règles de conception se montrent plus pertinentes et le moteur cognitif augmente la confiance qu'il porte en ses connaissances. La valeur de CG_5 diminue donc fortement pour atteindre la valeur de 1,16 décisions par expérience de conception en fin de simulation.

Les radios de référence procédant par force brute ne disposent d'aucune mémoire pour accumuler de l'expertise sur l'espace de conception. Elles ont donc besoin en moyenne d'un nombre sensiblement constant de décisions pour identifier la configuration optimale. Sur l'ensemble de la simulation, CG_5 prend une valeur de 6,80 décisions pour la première radio et de 4,52 décisions pour la seconde radio s'appuyant sur le graphe de robustesse pour stopper son expérimentation plus rapidement. L'introduction de connaissances expertes se traduit donc par un avantage certain pour l'exploration de l'espace de recherche (2,28 décisions économisées en moyenne par expérience de conception).

La figure 4.9 représente l'évolution du nombre de mauvaises décisions prises par expérience de conception (CG_6). Les courbes suivent la même évolution qu'à la figure précédente. L'introduction de connaissances expertes permet de limiter les risques d'exploration puisque le nombre de mauvaises décisions passe de 1,81 à 0,59 grâce au graphe de robustesse embarqué. Notre moteur cognitif termine la simulation avec un nombre moyen de mauvaises décisions bien plus faible (égal à 0,052 : 52 mauvaises décisions sont prises sur les 1000 derniers problèmes traités). Cet indicateur montre que le moteur cognitif est efficace dans sa recherche d'une configuration adaptée.

Analyse des expériences de conception menées par notre moteur cognitif

Pour analyser la réponse du moteur cognitif, nous introduisons les notations suivantes : O fait référence à une décision optimale, G à une décision correcte (good) et B à une mauvaise décision (bad). Chaque indicateur est alors associée à un motif d'évaluation caractéristique de la réponse du moteur cognitif :

- Pour l'indicateur CG_7 , le moteur cognitif n'a besoin que d'une seule décision pour trouver une configuration adaptée au problème. La configuration choisie est très souvent optimale (cf. fig. 4.6) ou proche de l'optimal (cf. fig. 4.7) donc le moteur cognitif a intérêt à exploiter ses connaissances après avoir accumulé suffisamment d'expérience. Le motif d'évaluation est alors égal à O ou G en fonction de l'optimalité de la décision.
- L'indicateur CG_8 témoigne aussi d'un comportement favorable pour l'utilisateur. L'exploration n'a pas d'impact si le moteur cognitif est conforté dans sa prédiction (aucune configuration n'est capable de délivrer le service correctement). L'incompatibilité du problème est validé après avoir testé la configuration la plus robuste donc le motif d'évaluation est égal à GO . Si l'expérience contredit sa prédiction, le moteur cognitif peut mettre à jour son modèle prédictif et ainsi éviter de répéter cette erreur. Le motif est alors égal à G^nOBO . La valeur de n peut être supérieure à zéro en début de simulation. Elle est égale à zéro quand l'espace commence à être bien caractérisé (i.e. la configuration la plus robuste est la configuration optimale).
- L'indicateur CG_{10} a un impact globalement positif puisque le moteur cognitif se rapproche du comportement optimal. Le motif d'évaluation correspond alors à G^nO si la contrainte

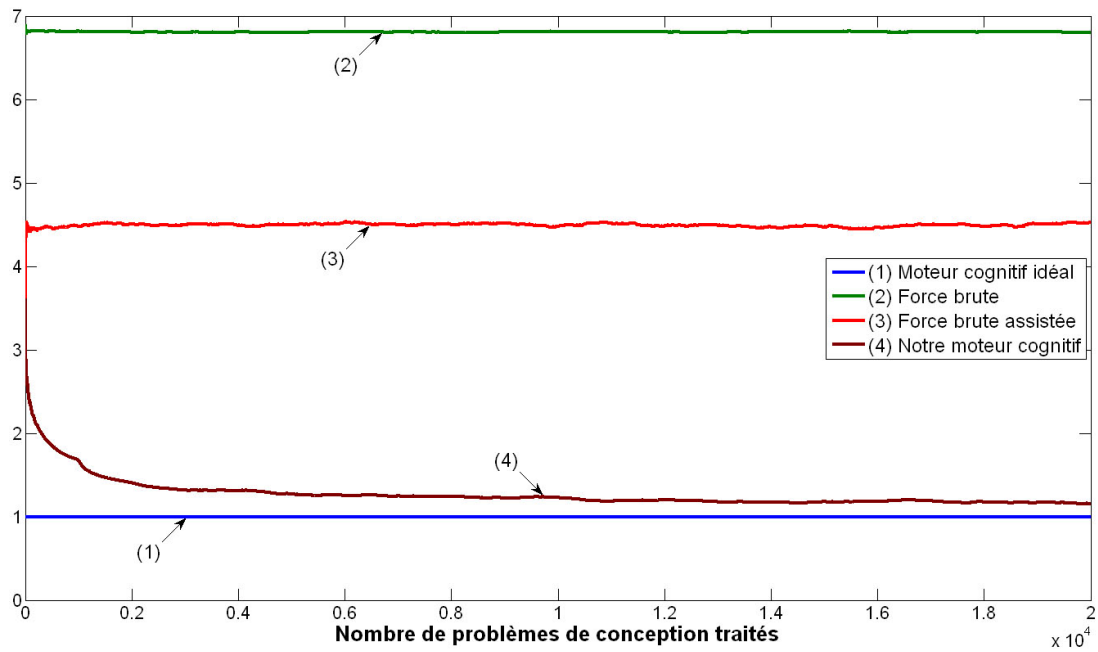


FIGURE 4.8 – Nombre moyenné de décisions prises par expérience de conception (CG_5)

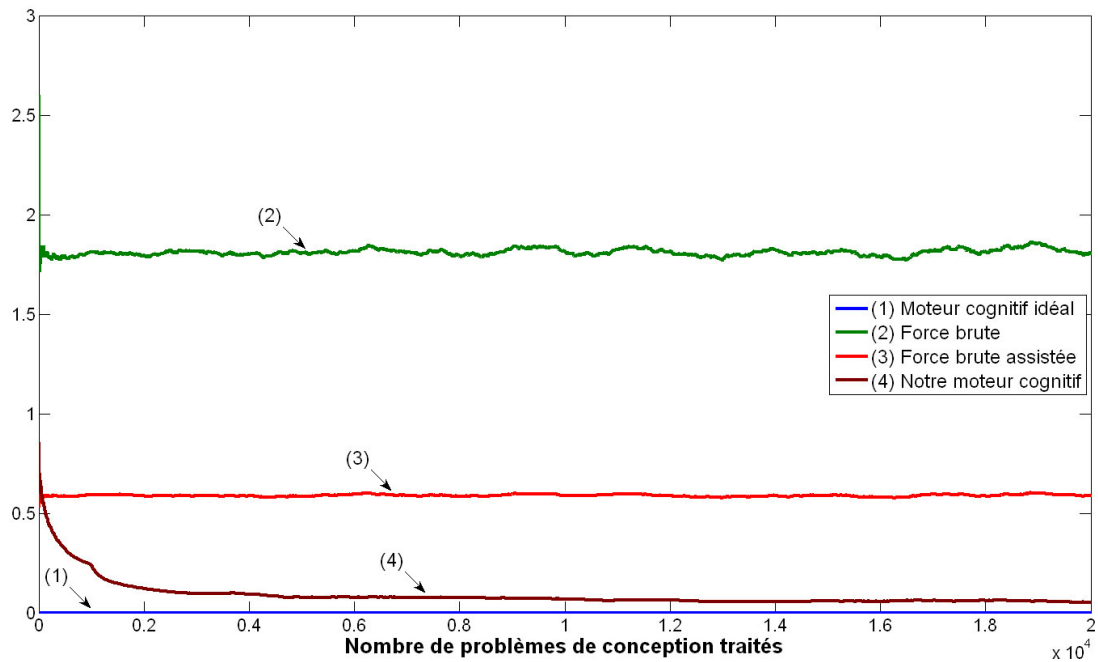


FIGURE 4.9 – Nombre moyenné de mauvaises décisions prises par expérience de conception (CG_6)

est vérifiée par toutes les configurations et à $G^n OBO$ sinon ($n \geq 1$). Par construction, le processus d'expérimentation introduit toujours une mauvaise décision (sauf pour des problèmes très faciles) puisqu'il continue son exploration jusqu'à identifier le HCR. Ce comportement est profitable en début de simulation, le moteur cognitif bénéficiant d'une marge de progression importante ($n \geq 1$). Mais le moteur cognitif a tendance à ne progresser que d'un seul rang lorsque l'espace commence à être bien caractérisé ($n = 1$). Il serait donc intéressant de limiter les risques d'expérimentation en contrôlant le dilemme exploration/exploitation après chaque décision au lieu d'attendre qu'une mauvaise décision ne soit prise. La même remarque est valable pour CG_8 .

- Le moteur cognitif est mis en défaut par ses connaissances avec l'indicateur CG_9 . Il profite alors de ses erreurs pour mettre à jour son modèle prédictif. Le motif d'évaluation est égal à $B^m G^n O$ avec $m \geq 1$ et $n \geq 0$. Les valeurs de m et n dépendent alors de la valeur de A_{def} et de la distance entre la prédiction et la valeur réelle du HCR. Le moteur cognitif ne se trompe plus que d'un seul rang lorsqu'il a acquis une certaine expérience sur son espace de conception. On aura donc $m = 1$ et $n = A_{def} - 1$ d'où notre choix du paramètre de comportement défensif ($A_{def} = 1$).
- L'indicateur CG_{11} a un impact globalement négatif, l'exploration entraînant une mauvaise décision sans aucun bénéfice pour le moteur cognitif (le modèle prédictif est déjà optimal pour les problèmes considérés). Le motif d'évaluation est alors égal à OBO .

La figure 4.10 illustre cette analyse affinée des réponses du moteur cognitif aux problèmes posés (CG_7 à CG_{11}). En début de simulation, le moteur cognitif est souvent induit en erreur par ses prédictions (cf. CG_9) et ses explorations offensives lui permettent de se rapprocher du comportement optimal (cf. CG_{10}) en dépit des risques pris (cf. CG_{11}). Le moteur cognitif limite ensuite progressivement son recours au processus d'expérimentation pour ne l'activer que pour 10,7% des cas en fin de simulation (cf. CG_7). Ses prédictions sont alors correctes dans la majorité des cas ($CG_9 = 0,7\%$) et ses tentatives d'exploration ne connaissent plus le même succès qu'en début de simulation ($CG_{10} = 0,33\%$). Le moteur cognitif apprend néanmoins à limiter les risques d'expérimentation grâce à l'équipe d'automates d'apprentissage gardant la trace des explorations échouées. L'indicateur CG_{11} diminue ainsi tout au long de la simulation pour atteindre une valeur de 4,23% pour les 1000 derniers problèmes. L'indicateur CG_8 intervient pour les problèmes sans solution lorsque le moteur cognitif décide de tester la configuration la plus robuste pour s'assurer de l'incompatibilité du problème. Il évolue moins vite que les autres car le seuil de vérification P_{CC} force le moteur cognitif à explorer tant qu'il ne dispose pas d'une assurance minimum sur la fiabilité de la règle appliquée.

Evolution de l'expérience du moteur cognitif sur son espace de conception

La figure 4.11 montre l'évolution de la probabilité d'exploration moyennée sur l'ensemble des automates d'apprentissage ($PSSE_1$). La probabilité diminue rapidement initialement puisqu'elle n'est plus qu'à 20% après 1000 problèmes traités. Le moteur cognitif dispose alors d'une caractérisation grossière mais efficace de son espace de conception. Il ajuste son modèle prédictif le reste de la simulation pour identifier les frontières de décision plus précisément. Les opportunités d'apprentissage se font plus rares car le moteur cognitif préfère exploiter ses connaissances pour limiter les risques d'expérimentation. La probabilité d'exploration diminue donc plus lentement jusqu'à atteindre la valeur de 7,9% en fin de simulation. La carte d'exploration présentée en annexe B.2 complète cette analyse du comportement de l'équipe d'automates d'apprentissage.

Les paramètres d'exploration R_V et a_p doivent être choisis avec soin pour optimiser le comportement du moteur cognitif. La probabilité moyenne d'exploration décroît plus rapidement

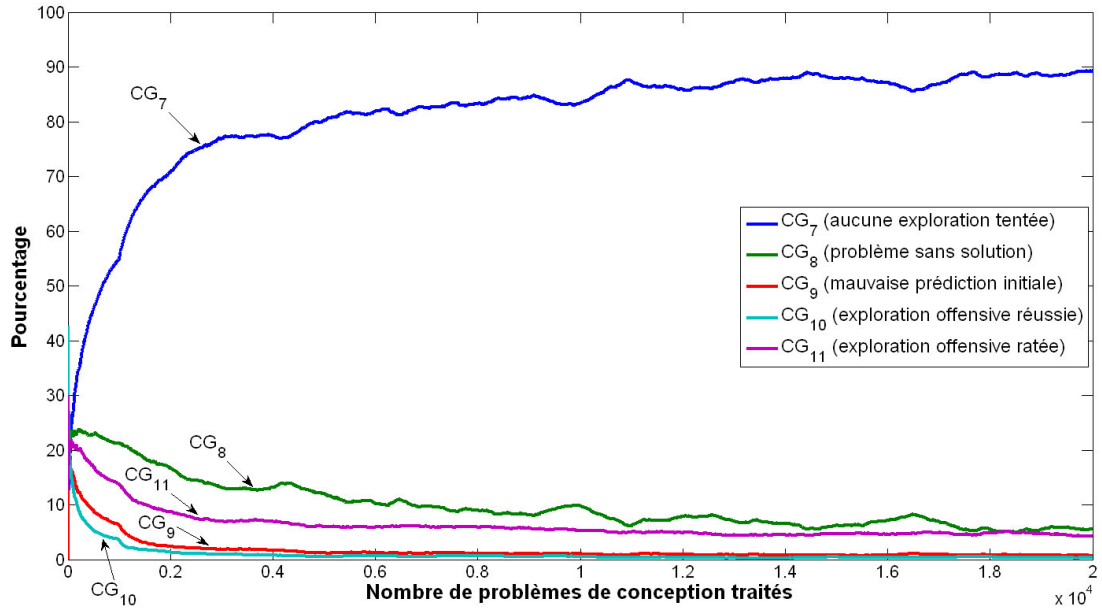


FIGURE 4.10 – Analyse de la réponse du moteur cognitif aux problèmes de conception posés (CG_{7-11})

pour une valeur de a_p élevée donc le moteur cognitif préfère limiter les risques d'expérimentation en favorisant l'exploitation de ses connaissances. Cette action a tendance à diminuer le nombre de mauvaises décisions liées aux échecs d'exploration. Le moteur cognitif s'éloigne en revanche du comportement optimal en étant moins offensif dans son exploration de l'espace de conception. Il peut alors passer à côté de possibilités d'amélioration, notamment au niveau des frontières de décision. Nous observons la même tendance pour R_V étant donné que la fréquence de mise à jour des automates augmente avec la valeur du rayon de voisinage.

Fiabilité du modèle prédictif

La figure 4.12 affiche l'évolution de la fiabilité du modèle prédictif et son impact pour la prise de décision. La qualité des prédictions du module CAP (FMP_2) ne cesse d'augmenter tout au long de la simulation pour atteindre une valeur de 92,8% en fin de simulation. L'espace commence à être bien caractérisé au bout de 1000 problèmes (82,3% de prédictions correctes). Le moteur cognitif affine ensuite son modèle prédictif tout en cherchant à limiter les risques d'expérimentation (donc les opportunités d'apprentissage). La progression est donc plus lente.

La courbe associée à FMP_1 permet d'illustrer la tolérance de notre moteur cognitif aux erreurs de prédiction. Elle est construite à partir des décisions que le moteur cognitif prendrait s'il devait exploiter ses connaissances pour tous les problèmes traités. Cette courbe est théorique car le modèle prédictif continue à évoluer avec les expérimentations réellement menées par le moteur cognitif. La qualité des décisions augmente donc avec la fiabilité croissante du modèle prédictif. Nous avons toujours $FMP_1 > FMP_2$ car le moteur cognitif peut prendre une décision optimale même si le module CAP se trompe dans sa prédiction. Il suffit en effet que le rang de la configuration optimale soit inférieur ou égal au HCR prédit. Cette tolérance explique les meilleures performances du moteur cognitif par rapport à celles du module CAP. Le gain relatif en fin de simulation est de 15,7% puisque le taux d'erreurs passe de 7,2% à 6,1%.

La figure 4.13 affiche la carte d'adéquation à différentes étapes de la simulation. Chaque

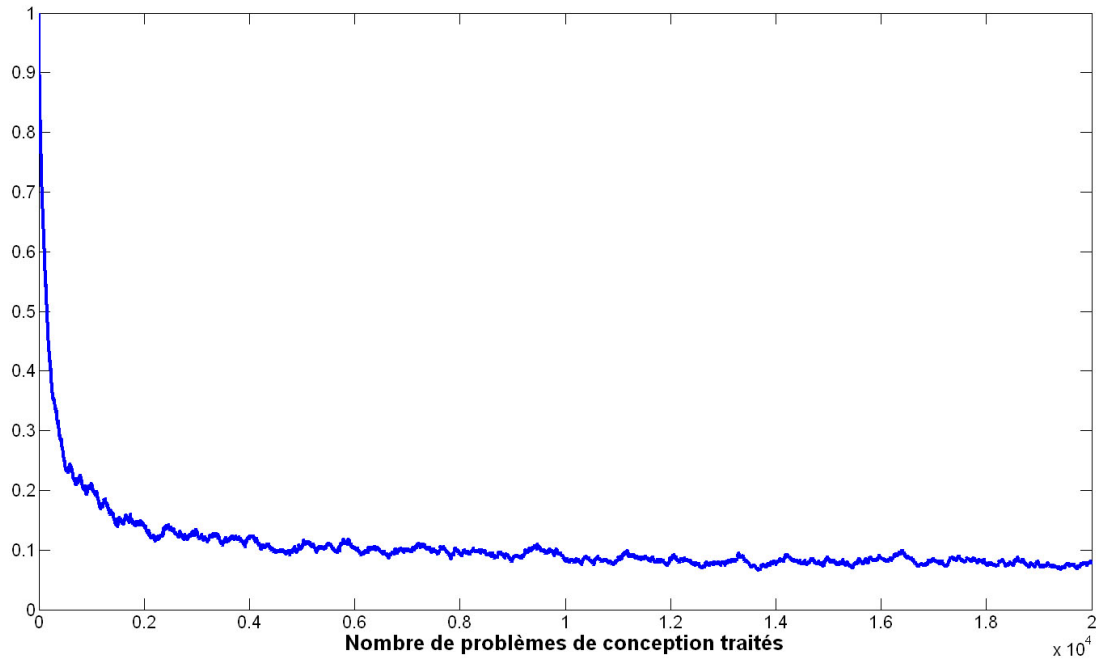


FIGURE 4.11 – Taux d'exploration moyenné sur l'ensemble des automates d'apprentissage ($PSSE_1$)

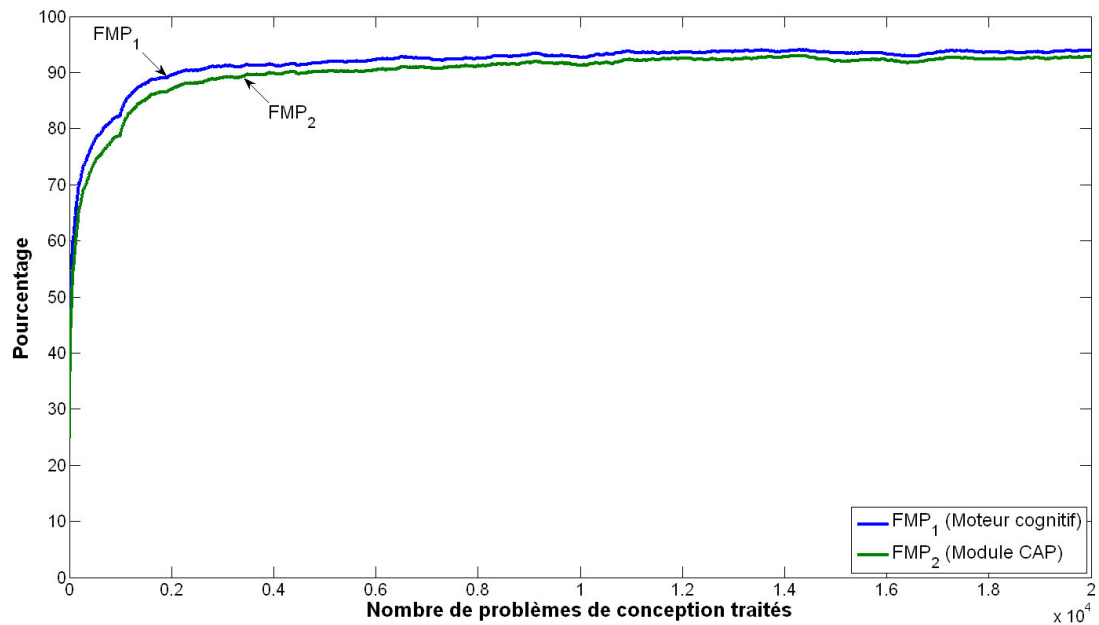


FIGURE 4.12 – Fiabilité des prédictions du module CAP (FMP_2) comparée à la qualité des décisions prises par le moteur cognitif (FMP_1)

cercle représente un cluster mémorisé par un nœud du système connexionniste. La position, le rayon et la couleur d'un cercle correspondent respectivement au centre du cluster, à son champ d'influence et à la classe associée (*i.e.* l'indice de la configuration compatible la moins robuste). Pour faciliter l'interprétation de la carte, nous avons repris le code de couleurs de la figure 3.4 et nous avons mis les frontières de décision en évidence.

Le moteur cognitif apprend progressivement à caractériser son espace de conception en interagissant avec son environnement par une procédure d'essais/erreurs. Le module CAP intègre les résultats d'expérimentation afin d'identifier les frontières de décision aussi précisément que possible. Sa progression s'observe facilement sur la figure 4.13.

La figure 4.13 permet aussi d'interpréter les réponses du moteur cognitif. Le moteur cognitif a tendance à généraliser brutalement tant qu'il n'est pas mis en défaut par l'expérimentation. Ses tentatives d'exploration sont encore peu nombreuses en début de simulation donc les règles ont tendance à s'étendre au-delà de leur région de classification. La prédiction devient mauvaise si la règle activée est associée à une configuration de rang supérieur au HCR. Elle est sous-optimale si la règle classe le problème vers une configuration de rang inférieur au HCR. La même logique s'applique lorsqu'aucune règle n'est activée. Le processus d'inférence du système connexionniste analyse alors les *M-of-N* règles les plus proches du problème traité afin d'identifier la classe la plus probable.

La couverture de l'espace de classification s'améliore avec l'expérience du moteur cognitif donc la fiabilité du modèle prédictif augmente progressivement. Après 2000 problèmes, le modèle est optimal dans la majorité des cas. Il est sous-optimal pour un faible nombre de problèmes proches des frontières de décision. La prédiction est néanmoins très satisfaisante car le module CAP ne se trompe que d'un seul rang dans l'échelle de performance. Cette observation confirme donc le besoin d'intégrer des métaconnaissances pour stopper l'exploration offensive après la première tentative réussie comme nous l'avons évoqué précédemment. Le nombre d'erreurs de prédiction continue à diminuer tout au long de la simulation, le moteur cognitif profitant de nouvelles expériences pour affiner son modèle prédictif.

Le paramètre R_{\min} a une influence importante sur le comportement du moteur cognitif. Une valeur trop grande s'accompagne d'un grand nombre d'erreurs de prédiction. Les clusters ne sont pas définis avec une granularité assez fine donc le système évolutif ne peut pas identifier les frontières de décision avec précision. Une valeur trop faible permet une caractérisation plus fine de l'espace de classification aux dépens d'un nombre de règles (donc d'une complexité) plus élevé. Le choix de R_{\min} requiert donc une bonne connaissance *a priori* de l'espace de conception. Cette contrainte forte sur la configuration du moteur cognitif peut cependant être réduite de plusieurs manières (*e.g.* adaptation en ligne des paramètres, mise en place d'une procédure d'agrégation des nœuds du système connexionniste).²⁵ Le paramètre R_{\max} influe sur le début de simulation en évitant une généralisation trop brutale pour des problèmes éloignés (*cf.* sec. 4.2.4). Son influence diminue rapidement lorsque la densité des règles augmente car l'expansion d'un nœud se trouve toujours bornée par ses voisins de classe différente.

Évaluation qualitative de la complexité du moteur cognitif

La figure 4.14 montre l'évolution du nombre de règles dans la base de connaissance développée par le module CAP. La base de connaissance croît rapidement en début de simulation pour mémoriser les règles déduites des expérimentations. Les phases d'exploration réussies se font plus rares au fur et à mesure que le moteur cognitif acquiert de l'expérience sur son espace de

25. Nous revenons sur ces améliorations au chapitre 6.

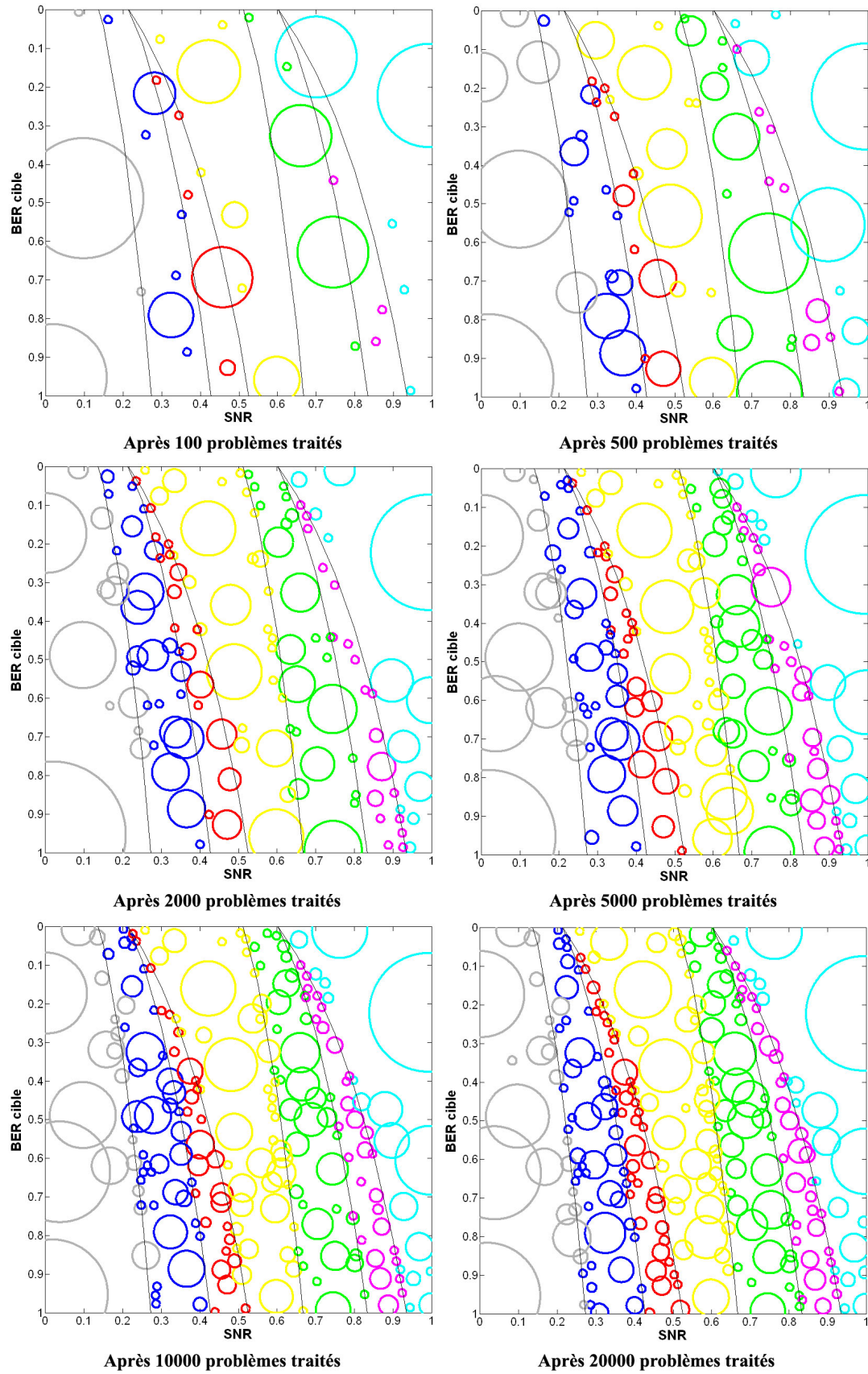


FIGURE 4.13 – Evolution de la carte d'adéquation avec l'expérience du moteur cognitif (FMP_3)

conception. Le nombre de règles augmente donc plus lentement pour atteindre la valeur de 255,8 après 20000 problèmes traités (la valeur est moyennée sur 10 simulations).²⁶

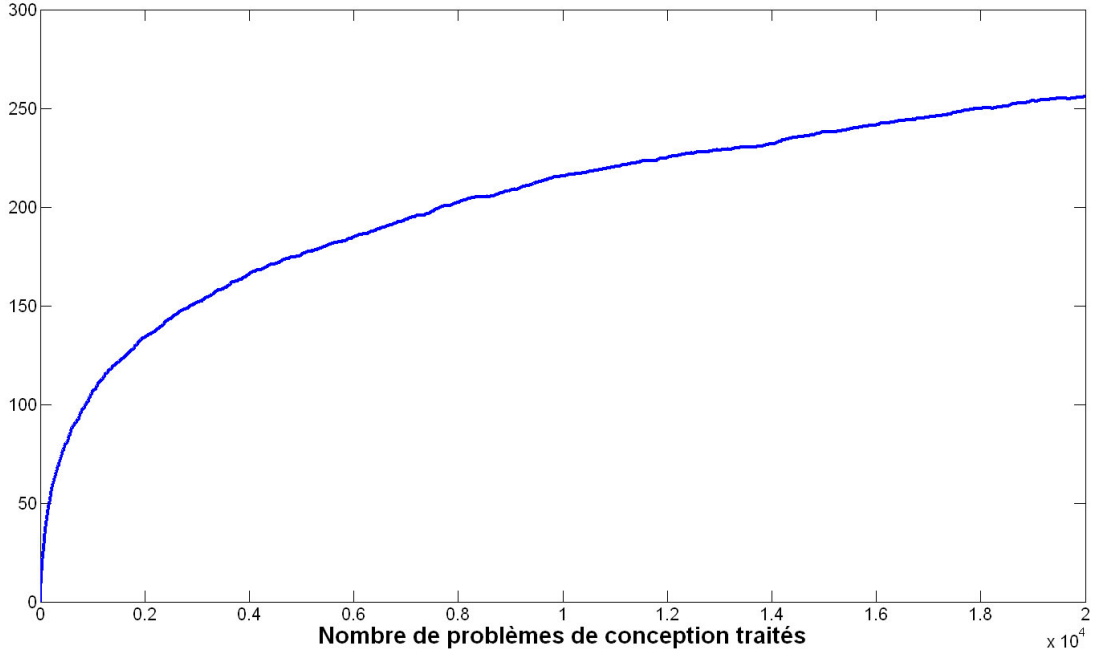


FIGURE 4.14 – Evolution du nombre de règles à l'intérieur de la base de connaissances (CMP_1)

La complexité du processus d'analyse prédictive croît linéairement avec le nombre de règles créées. Dans l'état actuel des mécanismes, le système connexionniste doit parcourir l'intégralité de sa base de connaissance avant de retourner le HCR prédit. Il estime l'intérêt de chaque règle vis-à-vis du problème traité à partir d'un calcul de distance très léger en ressources algorithmiques (*cf.* équation 4.1). La multiplication des calculs peut néanmoins s'avérer handicapante pour la complexité du moteur cognitif et sa réactivité. Le nombre de règles créées n'est pas contrôlable en pratique puisqu'il dépend de la taille de l'espace de conception, du nombre de frontières de décision et de la superficie des régions de classification.²⁷ Les mécanismes actuels ne conviennent donc pas pour des études de cas complexes impliquant une grande flexibilité de l'équipement. Comme nous l'exposons au chapitre 6, de nombreuses solutions sont envisageables pour remédier à ce problème. Les bases de connaissance peuvent être parcourues plus efficacement et le moteur cognitif peut exploiter des mécanismes avancés de gestion de la complexité.

La complexité du processus de décision est maîtrisée à travers trois facteurs. Premièrement, la méthode TOPSIS est une méthode relativement simple reposant sur des calculs de distance et des comparaisons (*cf.* sec.4.2.2). Deuxièmement, la matrice de décision peut être pré-remplie grâce aux informations stockées dans le profil de l'équipement. Troisièmement, le processus de décision évalue seulement les configurations supposées compatibles avec le service. Le système

26. Ce nombre est caractéristique de l'espace de conception exploré dans son intégralité. En pratique, la contrainte de QoS n'est pas censée prendre des valeurs continues mais des valeurs discrètes typiques de l'application sélectionnée. Le nombre de règles nécessaires sera donc considérablement réduit.

27. Il dépend aussi des paramètres choisis, notamment de R_{\min} et des paramètres d'exploration. Le nombre de règles augmente lorsque le système connexionniste évolue avec une granularité plus fine (valeur faible de R_{\min}). Les opportunités d'apprentissage (donc le nombre de règles créées) diminuent si le moteur cognitif a tendance à favoriser l'exploitation plus rapidement (a_p élevé, R_V élevé).

de notation ne gaspille donc pas de ressources de calcul pour les configurations écartées par le modèle prédictif.

Le processus d'expérimentation navigue sur l'échelle de performance pour réagir aux retours de l'environnement en suivant une logique simple de progression ou de régression. Sa complexité est donc extrêmement faible.

Grâce au processus d'analyse prédictive, le processus d'apprentissage connaît l'ensemble des règles activées par le problème ainsi que l'ensemble des règles impliquées dans son voisinage. Il met à jour les automates d'apprentissage voisins à l'aide d'une procédure très légère en ressources de calcul (*cf.* sec. 4.2.4). Il actualise ensuite la base de connaissance si nécessaire. Cette phase peut être plus ou moins complexe en fonction de l'heuristique appliquée. La création d'un nœud est une opération simple tout comme la réduction du champ d'influence d'une règle erronée. La procédure de généralisation d'une règle est par contre beaucoup plus complexe. Le processus doit en effet calculer la distance entre la règle généralisée et les règles de classe différente avant de pouvoir valider l'augmentation de son champ d'influence. La complexité du processus d'apprentissage dépend donc de la taille de la base de connaissance dans ce cas. Comme pour le processus d'analyse prédictive, cette dépendance peut être considérablement réduite grâce aux solutions exposées au chapitre 6.

Impact du moteur cognitif sur les indicateurs de performance de la radio

Les bénéfices de la reconfiguration dynamique sont mis en valeur dans le tableau 4.3. Le tableau recense les valeurs des indicateurs de performance BRD_{1-3} ²⁸ moyennées sur l'ensemble de la simulation pour :

- les radios non flexibles associées aux configurations considérées dans le scénario (C_1 à C_6),
- des radios capables de se reconfigurer sous les ordres d'un moteur cognitif embarqué.

	Radios non flexibles						Radios flexibles	
	C_1	C_2	C_3	C_4	C_5	C_6	Moteur cognitif idéal	Moteur cognitif proposé
BRD_1	78,2	65,7	59,5	39,6	25,9	19,4	78,2	77,7
BRD_2	0,57	0,62	0,59	0,86	0,73	0,58	1,26	1,24
BRD_3	476	8	1	476	8	1	82,7	84,7

TABLE 4.3 – Comparaison des indicateurs de performance BRD_{1-3} pour des radios non flexibles et flexibles

Le tableau montre que notre moteur cognitif assure une gestion optimisée des ressources tout en garantissant la QoS requise.²⁹ Le tableau témoigne aussi des bonnes performances de notre moteur cognitif par comparaison à sa version idéalisée.

Le BER cible est atteint dans 77,7% des cas au lieu de 78,2% avec la configuration la plus robuste (C_1). La différence est très faible. On pourrait la réduire encore plus en diminuant P_{CC} .

La mesure du taux d'erreurs ne tient pas compte du temps passé à rechercher une configuration suffisamment performante. Les mauvaises décisions prises lors de l'exploration en ligne

^{28.} Pour rappel, BRD_1 est associé au maintien de la QoS, BRD_2 au débit moyen de la communication et BRD_3 à la complexité algorithmique des traitements appliqués au signal.

^{29.} Les performances de la radio sont normalement mesurées en continu donc les chiffres donnés dans le tableau ne sont qu'indicatifs (le moteur cognitif est testé suivant une approche statique).

viennent dégrader la qualité de service perçue par l'utilisateur. Cette dégradation est ponctuelle. Elle est plus ou moins fréquente en fonction :

- *de la vitesse d'évolution du contexte* : Une configuration reste performante moins longtemps si l'environnement varie rapidement. La phase d'exploration en ligne a alors un impact non négligeable sur la durée de la communication. Son impact est beaucoup moins grand si le moteur cognitif n'a pas besoin d'adapter le comportement de la radio régulièrement.
- *du nombre d'expériences accumulées* : Le moteur cognitif ne fait plus appel au processus d'expérimentation après avoir accumulé suffisamment d'expérience sur la région visitée. Le moteur cognitif est donc particulièrement performant vis-à-vis des situations fréquemment rencontrées par l'utilisateur tout en gardant ses capacités d'adaptation pour des situations plus inhabituelles.

La phase de recherche n'a pas toujours un impact désastreux sur la QoS. Le processus d'expérimentation prend de bonnes décisions lorsqu'il découvre une configuration compatible isolée (il se rapproche également du comportement optimal). Le taux d'erreurs augmente après une mauvaise décision sans être dégradé brutalement. Le processus d'expérimentation navigue avec prudence sur l'échelle de performance afin de minimiser les baisses de performances de la radio. De plus, une configuration incompatible avec le service n'est pas maintenue très longtemps, le moteur cognitif proposant une nouvelle configuration dès qu'il observe avoir pris une mauvaise décision.

La radio contrôlée par notre moteur cognitif est beaucoup plus satisfaisante du point de vue des objectifs secondaires comparé à C_1 . La complexité algorithmique des traitements appliqués est réduite de 476 opérations en moyenne à 84,7 et le débit d'information utile est plus que doublé. La complexité indiquée ne tient pas compte de la complexité additionnelle des mécanismes cognitifs qu'il serait nécessaire de quantifier en conditions d'utilisation. En pratique, le moteur cognitif a un impact énergétique beaucoup moins régulier que la radio. Le moteur cognitif peut se montrer plus complexe momentanément lors de la recherche d'une configuration adaptée et de l'apprentissage. Son coût algorithmique est par contre beaucoup plus faible tant que la configuration s'avère en adéquation avec le contexte. Conscients de cette différence, nous proposons un processus de surveillance très léger en ressources de calcul au chapitre 6.

Les indicateurs de performance ont été calculés pour une distribution homogène de problèmes de conception pris sur l'ensemble de l'espace de conception. Les bénéfices de la reconfiguration dynamique varient cependant avec la difficulté des problèmes rencontrés. Le moteur cognitif est capable d'optimiser pleinement l'ensemble des indicateurs de performance pour des problèmes faciles (*e.g.* SNR élevé, exigences de QoS faibles). Pour des problèmes plus difficiles, il cherche avant tout à délivrer la QoS requise avant de songer à optimiser les autres indicateurs de performance.

Lors de la simulation, le moteur cognitif est soumis à des problèmes générés aléatoirement. La politique d'optimisation est donc susceptible de varier d'un problème à l'autre. Nous avons relancé les simulations en ne considérant qu'une politique d'optimisation à la fois. Les résultats sont présentés dans le tableau 4.4.

Politique d'optimisation	OP_1	OP_2	OP_3
BRD_1	77,6	77,7	77,7
BRD_2	1,09	1,38	1,21
BRD_3	62,1	128,4	63,0

TABLE 4.4 – Influence de la politique d'optimisation sur les indicateurs de performance

Le moteur cognitif cherche en priorité à maintenir la QoS perçue par l'utilisateur. L'indicateur BRD_1 est donc sensiblement le même quelque soit la politique d'optimisation appliquée. Le débit d'information utile (BRD_2) est maximal lorsque le moteur cognitif respecte le mode multimédia associé à la politique d'optimisation OP_1 . Il est minimal lorsque l'accent est mis sur l'autonomie énergétique (OP_2). Cette tendance s'inverse lorsqu'on observe la complexité algorithmique moyenne des traitements appliqués (BRD_3). La politique d'optimisation OP_3 permet au moteur cognitif d'assurer un bon compromis entre les objectifs secondaires. Il est même préférable de la favoriser par rapport à OP_1 puisqu'elle conduit à une complexité moyenne similaire pour un débit plus grand. Ce résultat s'explique par la très faible complexité de décodage du premier schéma de transmission (7 opérations) comparée au second (475 opérations).

Mécanismes cognitifs pour la prise de décision autonome dans un espace de conception partiellement ordonné

Sommaire

5.1	Organisation structurelle du moteur cognitif généralisé	98
5.2	Extension des mécanismes cognitifs	99
5.2.1	Processus d'analyse prédictive	99
5.2.2	Processus d'expérimentation	101
5.2.3	Processus d'apprentissage	108
5.2.4	Algorithme de supervision	111
5.3	Evaluation du moteur cognitif	113
5.3.1	Présentation d'indicateurs de performance supplémentaires	113
5.3.2	Résultats	115

Les mécanismes cognitifs décrits au chapitre précédent ne sont plus aussi performants lorsque l'espace de conception n'est que partiellement ordonné. Les connaissances embarquées ne permettent plus d'identifier une échelle de performance fiable pour chaque état du canal radio. Les erreurs de classement induisent le processus d'expérimentation en erreur qui a tendance à adopter un comportement excessivement défensif à chaque fois qu'une mauvaise décision est prise. Le moteur cognitif apprend donc un comportement largement sous-optimal ce qui limite l'intérêt des mécanismes proposés.

Des améliorations sont donc nécessaires pour que le moteur cognitif évolue avec succès dans un espace de conception moins bien caractérisé. Le moteur cognitif doit apprendre à prédire une échelle de performance fiable pour la prise de décision quelque soit les conditions de propagation. La structuration progressive de l'espace de conception a alors besoin d'être guidée par un processus d'expérimentation plus évolué. Le moteur cognitif doit notamment pouvoir explorer les configurations de rang supérieur au HCR afin d'identifier d'éventuelles erreurs de classement. Il doit aussi cesser de toujours chercher à régresser le long de l'échelle de performance en cas de mauvaises décisions issues de l'exploitation des connaissances.

Dans ce chapitre, nous détaillons les améliorations que nous avons proposées pour traiter le cas plus général d'un espace de conception partiellement ordonné. Nous commençons par présenter la nouvelle organisation du moteur cognitif puis nous détaillons les modifications apportées aux

processus revisités (analyse prédictive, expérimentation, apprentissage) ainsi qu'à l'algorithme de supervision. Le moteur cognitif muni des extensions proposées est ensuite évalué sur une nouvelle étude de cas plus complexe.

5.1 Organisation structurelle du moteur cognitif généralisé

L'organisation du moteur cognitif généralisé est illustrée à la figure 5.1.³⁰ Les améliorations portent sur quatre points que nous présentons l'un après l'autre.

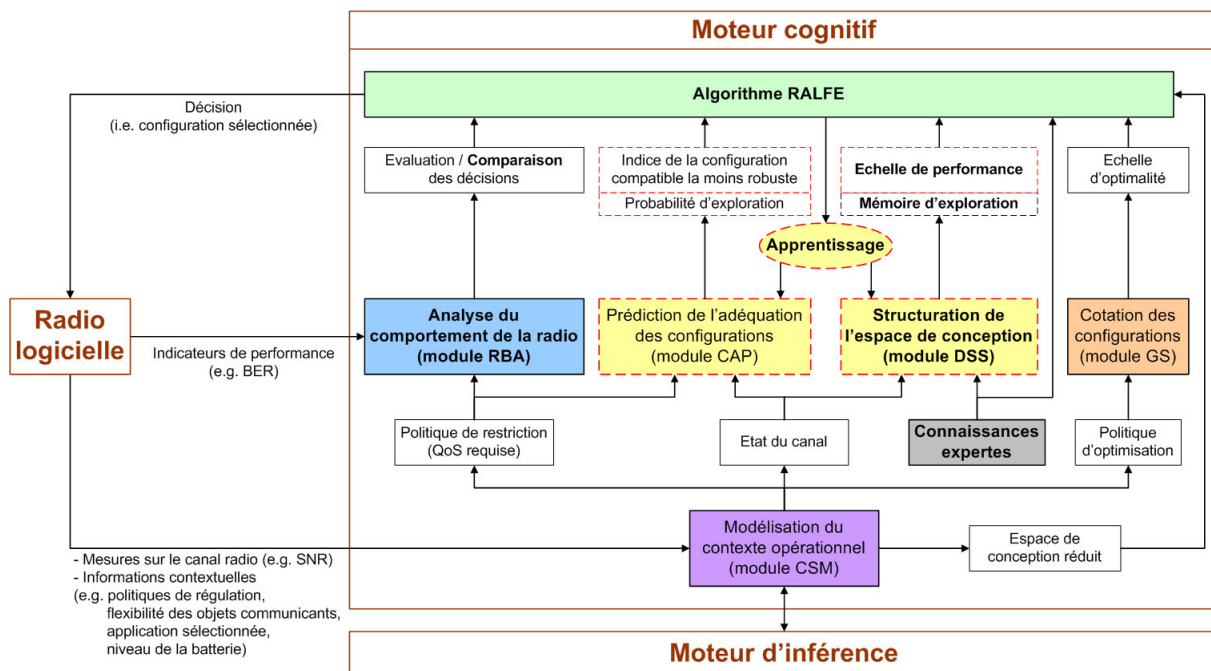


FIGURE 5.1 – Organisation structurelle du moteur cognitif généralisé
Les améliorations concernent les modules/informations mises en valeur en gras.

Nous avons introduit le module DSS (*Design Space Structurer*) pour structurer progressivement l'espace de conception du moteur cognitif. Ce module s'appuie sur un système connexionniste évolutif pour prédire l'échelle de performance s'appliquant à l'environnement radio modélisé. La qualité de ses prédictions augmente par apprentissage au fur et à mesure que le moteur cognitif prend de l'expérience sur son espace de conception. Le module peut aussi compter sur les connaissances expertes pour limiter les erreurs de prédictions puisqu'il s'assure toujours de produire une échelle compatible avec le graphe de robustesse embarqué. Le module profite enfin du partitionnement local de l'espace de conception (réalisé par son système connexionniste) pour garder une trace des explorations passées et ainsi guider plus efficacement les expérimentations futures.

La deuxième modification concerne l'expérimentation. Dans un espace de conception totalement ordonné, l'algorithme RALFE base ses expérimentations uniquement sur l'échelle de performance obtenue à partir de connaissances expertes. Si les relations d'ordre ne sont plus

30. Comme à la section 4.1, nous distinguons les informations en fonction de leur fiabilité pour la prise de décision (trait plein/pointillé). Les modules dotés de capacités d'apprentissage (pointillé) sont aussi différenciés des autres (trait plein).

totales mais partielles, la pertinence du plan d'expérience dépend de la qualité des prédictions du module DSS au voisinage du problème de conception traité. Néanmoins, l'algorithme peut toujours compter sur des connaissances expertes pour inférer sur les tentatives et ainsi réagir plus intelligemment aux retours de l'environnement.

La troisième modification est liée au module RBA. Le module ne se contente plus seulement d'évaluer la qualité des décisions prises (contrainte de performance vérifiée ou non). Il compare aussi les performances des configurations tentées afin de repérer d'éventuelles erreurs de classement dans l'échelle de performance prédite.

La dernière modification porte sur l'apprentissage. Les informations utilisées pour l'entraînement des systèmes connexionnistes ne sont pas toujours optimales, le moteur cognitif n'étant jamais certain d'avoir complètement caractérisé la région visitée. La stratégie locale de prise de décision peut donc s'avérer sous-optimale dans un premier temps. Confronté à un problème similaire, le moteur cognitif peut se rendre compte de sa sous-optimalité par l'expérimentation. Il peut alors identifier une meilleure stratégie le rapprochant encore plus du comportement idéal. Les systèmes prédictifs doivent cependant être capables d'oublier les associations sous-optimales. Nous avons donc introduit des mécanismes de désapprentissage à cet effet.

5.2 Extension des mécanismes cognitifs

Les nouveaux mécanismes proposés viennent compléter l'organisation fonctionnelle du moteur cognitif décrit au chapitre 4. Les processus d'analyse prédictive et d'apprentissage ont été modifiés afin de tenir compte de la nouvelle méthode d'identification de l'échelle de performance (évolution progressive d'un modèle prédictif). Le processus d'expérimentation a été revisité en grande partie alors que le processus de décision n'a subi aucun changement. L'algorithme de supervision utilise de nouvelles règles pour gérer les passages du processus de décision vers le processus d'expérimentation.

5.2.1 Processus d'analyse prédictive

Le processus d'analyse prédictive s'appuie désormais sur le modèle prédictif développé par le module DSS pour établir l'échelle de performance en fonction des conditions de propagation. Ce modèle est mémorisé dans la structure d'un système connexionniste évolutif similaire à celui utilisé par le module CAP (modèle ECF). Son fonctionnement a été décrit à la section 4.2.1. Nous avons simplement adapté l'heuristique d'initialisation du système d'inférence en gardant le même principe de précaution : si la base de connaissance ne contient aucune règle, le module DSS retourne l'échelle de performance obtenue en appliquant la version prudente de l'algorithme 1 au graphe de robustesse embarqué.

Pour améliorer l'efficacité du processus d'expérimentation, nous avons associé une mémoire d'exploration \mathbf{ME}_k à chacune des règles Cl_k du module DSS. Cette mémoire permet de garder une trace des expériences de conception passées. Elle se matérialise sous la forme d'un vecteur contenant autant d'éléments que de configurations disponibles.³¹ L'élément ME_k^m correspond au poids d'exploration associé à la configuration C_m . Une configuration a plus de chance d'être testée par le processus d'expérimentation si son poids d'exploration prend une valeur faible (le poids minimum est égal à 1). Une valeur forte du poids d'exploration implique au contraire que

31. Cette méthode peut devenir très coûteuse en ressources mémoire pour de très grands espaces de conception. Ce point est discuté dans le chapitre 6 où nous suggérons des améliorations comme perspectives pour la suite de ces travaux.

la configuration ne s'est pas montrée suffisamment performante lors des tentatives précédentes donc il est préférable de l'éviter plutôt que de la tester à nouveau.

La mémoire d'exploration est calculée à partir d'un raisonnement semblable à celui appliqué lors de l'estimation de la probabilité locale d'exploration (eq. 5.1). Le module DSS récupère les mémoires d'exploration associées aux règles voisines du problème de conception (*i.e.* se trouvant à une distance $d_k < R_V$) puis il effectue une moyenne pondérée des différents vecteurs afin d'identifier une mémoire d'exploration représentative de la région de l'espace de conception considérée.

$$\mathbf{ME} = \frac{\sum_{k \in V} w_k \mathbf{ME}_k}{\sum_{k \in V} w_k} \text{ avec } w_k = \exp\left(-\frac{d_k}{R_V}\right) \quad (5.1)$$

Une fois l'échelle de performance identifiée et la mémoire d'exploration calculée, l'analyse prédictive se poursuit par la prédiction de la configuration compatible la moins robuste et par l'estimation de la probabilité locale d'exploration. Le fonctionnement du module CAP n'est pas modifié (*cf.* sec. 4.2.1).

La figure 5.2 illustre l'impact des prédictions (CAP et DSS) sur la qualité des décisions prises. L'exemple proposé implique un espace de conception partiellement ordonné contenant 10 configurations (C_a à C_j). Pour le problème de conception posé, les configurations sont en gras si elles sont compatibles avec le service et en italique si elles ne sont pas adéquates.

Le moteur cognitif prend une décision incertaine s'il n'a pas su identifier parfaitement l'ensemble des configurations compatibles avec le service. Ses connaissances sont sous-optimales si le modèle exclut des configurations pourtant compatibles avec le service. Elles sont erronées si le modèle intègre des configurations trop peu performantes dans son ensemble de configurations compatibles.

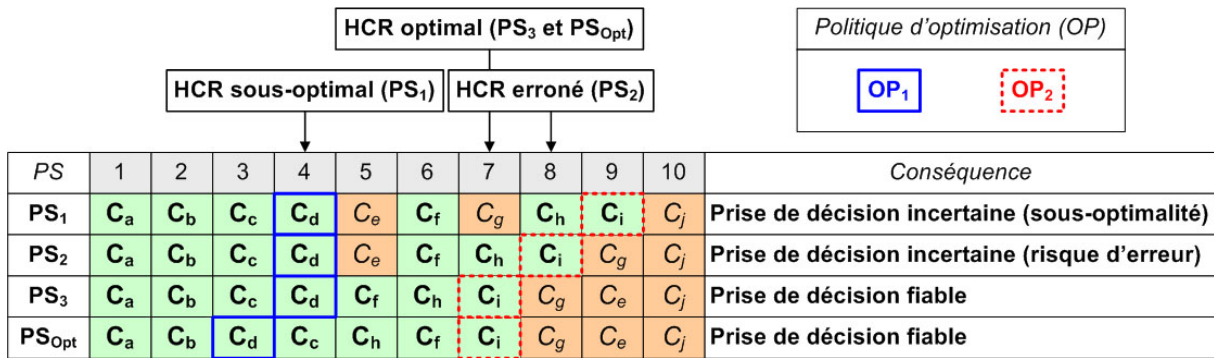


FIGURE 5.2 – Influence de la fiabilité des prédictions du module DSS sur la qualité de la décision prise

PS_i : Echelle de performance prédite par le module DSS, PS_{Opt} : Relations d'ordre réelles entre les alternatives

Le premier cas d'incertitude est observé pour l'échelle de performance PS_1 lorsque le module CAP identifie C_d comme la configuration compatible la moins robuste. Le moteur peut néanmoins prendre une décision optimale tant que la configuration la plus satisfaisante appartient à l'ensemble des configurations prédites comme compatibles (C_d pour OP_1). La décision est sous-optimale (mais correct) dans le cas contraire (pour OP_2 , le processus de décision choisit une configuration moins satisfaisante que C_i parmi C_a , C_b , C_c et C_d).

Le second cas d'incertitude est observé pour l'échelle de performance PS_2 lorsque le module CAP identifie C_i comme la configuration compatible la moins robuste. Les configurations optimales pour OP_1 et OP_2 font partie des configurations prédites comme compatibles donc le

moteur cognitif aboutit forcément à une décision optimale. Par contre, il peut être induit en erreur lors du processus de décision si la configuration C_e est plus satisfaisante que C_d/C_i vis-à-vis de la politique d'optimisation appliquée (OP_1/OP_2). Le moteur se rend compte de son erreur après l'avoir testé. Il peut alors mettre à jour son modèle prédictif pour l'exclure de l'ensemble des configurations compatibles.

Le premier type d'incertitude impose une démarche active d'exploration pour aller chercher les configurations compatibles isolées. Ce type d'incertitude est donc plus difficile et plus risqué à traiter. Le second type d'incertitude est préférable en comparaison car le moteur cognitif a juste à réagir à sa mauvaise décision pour corriger son erreur de classement (démarche passive d'exploration).

La dernière échelle de performance PS_3 assure une prise de décision fiable lorsque le module CAP identifie C_i comme la configuration compatible la moins robuste. Les relations d'ordre prédites ne sont pourtant pas exactes (*cf.* PS_{Opt}) mais les erreurs de classement n'ont aucun impact sur la prise de décision (tolérance aux imprécisions). Cependant, il reste préférable d'ordonner les configurations aussi précisément que possible afin de pouvoir traiter un panel plus large de problèmes de conception pour une même échelle de performance (et ainsi limiter la complexité du module DSS). Une échelle de performance est prédite pour un état du canal radio donné. Le moteur cognitif travaille avec la portion basse de l'échelle de performance si le service est exigeant en QoS et avec la portion haute pour des problèmes plus simples. Les erreurs de classement sans impact pour des HCR élevés peuvent alors conduire à un modèle sous optimal ou erroné pour des HCR plus faibles.

5.2.2 Processus d'expérimentation

Le processus d'expérimentation a désormais deux objectifs : structurer l'espace de conception et découvrir la configuration compatible la moins robuste. Il est donc nécessaire de réaliser une exploration plus poussée de l'espace de conception afin d'identifier correctement toutes les configurations compatibles avec le problème. Le schéma réactif de la figure 4.2 laisse donc place au schéma plus avancé de la figure 5.3.

Le moteur cognitif cherche toujours à identifier une configuration vérifiant la contrainte de QoS (s'il en existe une) avant de passer à l'exploitation. Lorsqu'il en a trouvé une, il explore de nouvelles alternatives afin de rechercher les configurations victimes d'erreurs de classement parmi celles compatibles avec le service. Il maintient son exploration offensive tant qu'elle n'impacte pas l'expérience de l'utilisateur (*i.e.* tant que la configuration sélectionnée vérifie la contrainte de performance). Il arrête l'expérimentation en cas de mauvaises décisions répétées afin de laisser place à l'exploitation.

Les principales différences entre les deux schémas réactifs sont détaillées et commentées ci-dessous.

Mise à jour des informations d'exploration

Le moteur cognitif ne peut plus avoir une confiance absolue dans l'échelle de performance donc il maintient un tableau d'exploration à jour lorsqu'il expérimente. Ce tableau d'exploration **TE** est représenté par un vecteur de longueur égale au nombre de configurations disponibles. L'élément TE^m mémorise l'état de connaissance du moteur cognitif quant à l'adéquation éventuelle de la configuration C_m vis-à-vis du problème posé. Face à un nouveau problème, le tableau d'exploration est initialisé à l'état *I* (Indéterminé) pour toutes les configurations. Le tableau est ensuite mis à jour à chaque tentative du moteur cognitif. L'état *VC* (Violated Constraint)

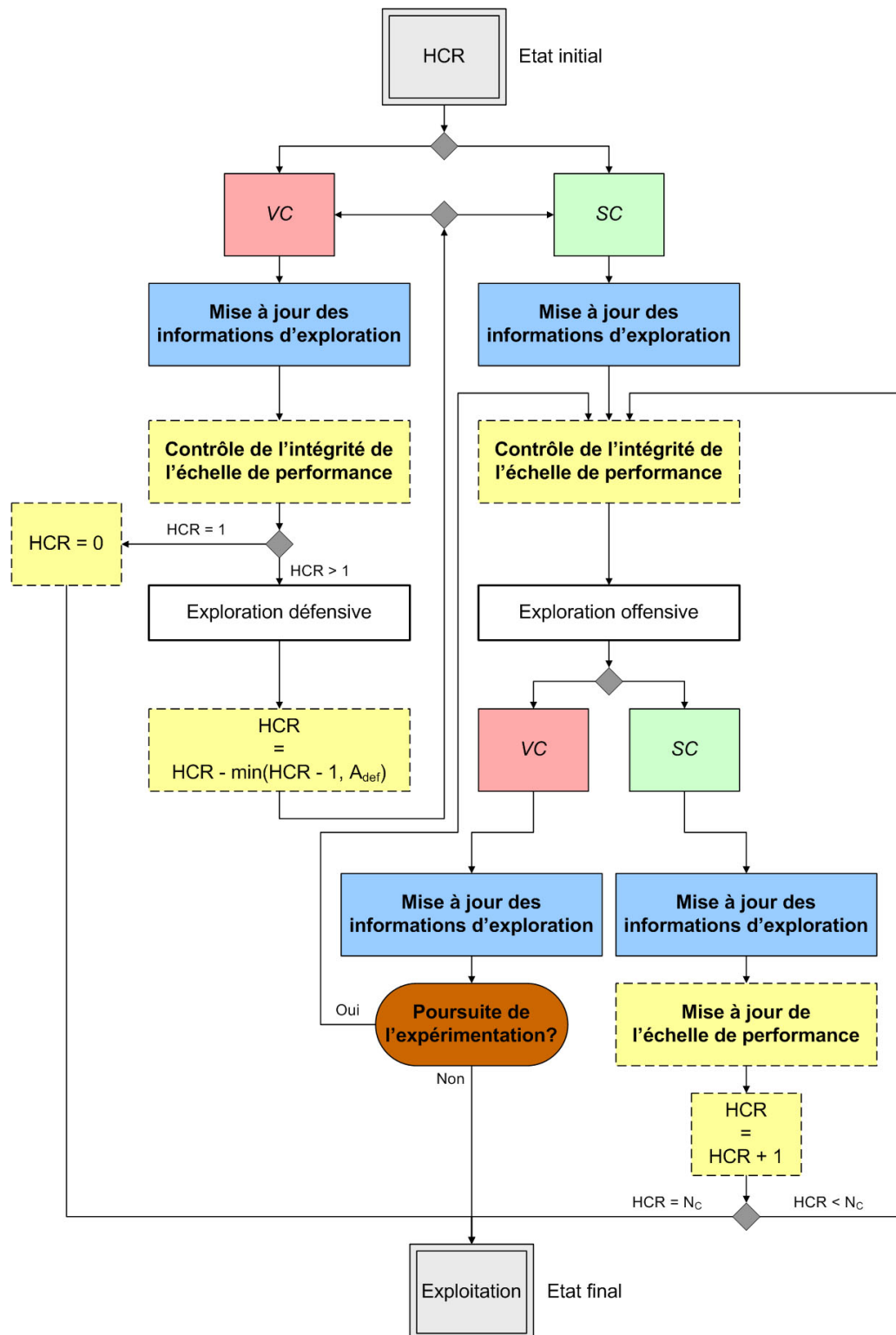


FIGURE 5.3 – Déroulement du processus d'expérimentation pour un espace de conception partiellement ordonné

indique que la configuration correspondante n'atteint pas les exigences de QoS. L'état *SC* (Satisfied Constraint) signale au contraire que la configuration est suffisamment performante pour délivrer le service correctement. Très souvent, la mise à jour du tableau d'exploration ne se limite pas seulement à l'élément associé à la configuration testée. Le moteur cognitif dispose en effet de connaissances expertes sur les relations d'ordre entre les configurations. Ces connaissances peuvent être exploitées par le moteur cognitif pour inférer sur les performances d'autres configurations :

- Si la configuration testée C_i donne lieu à une mauvaise décision, le graphe de robustesse est parcouru afin d'identifier toutes les configurations moins robustes que C_i (*i.e.* l'ensemble de ses configurations enfants) et les éléments correspondants sont mis à l'état *VC*.
- Si la configuration testée C_i donne lieu à une décision correcte (voire optimale), le graphe de robustesse est parcouru afin d'identifier toutes les configurations plus robustes que C_i (*i.e.* l'ensemble de ses configurations parents) et les éléments correspondants sont mis à l'état *SC*.

Le processus d'expérimentation ne cherche donc jamais à tester une configuration moins robuste qu'une configuration connue comme trop peu performante (diminution des risques d'exploration).

En plus du tableau d'exploration, le moteur cognitif met à jour un vecteur **CME** pour corriger les mémoires d'exploration dans le voisinage du problème de conception traité. Les éléments CME^m sont nuls initialement pour l'ensemble des configurations disponibles. Ils sont ensuite décrémentés/incrémentés à chaque fois que la configuration C_m est identifiée comme une bonne/mauvaise décision (*SC/VC*). Ces conclusions peuvent être obtenues par une expérimentation directe ou par inférence. Les compteurs CME^m peuvent donc être actualisés plusieurs fois pour une même expérience de conception. Une configuration reconnue plusieurs fois comme inadéquate n'apparaît pas comme une configuration intéressante pour les problèmes de conception similaires donc il est logique d'augmenter son poids d'exploration d'autant plus.

Illustrons la mise à jour des informations d'exploration à l'aide d'un exemple. Supposons que le moteur cognitif dispose du graphe de robustesse défini pour les 27 configurations du scénario de reconfiguration. Ce graphe, introduit à la figure 3.10, permet de structurer l'espace de conception de manière partielle mais fiable. Les deux vecteurs d'exploration sont réinitialisés pour chaque nouveau problème en faisant $TE^m = I$ et $CME^m = 0$ pour $m = 1, \dots, 27$. Supposons, de plus, que le moteur cognitif décide de tester la configuration C_{14} . Cette décision peut être prise dans deux situations : soit C_{14} correspond à la configuration optimale compte tenu des connaissances actuelles du moteur cognitif (exploitation), soit elle est associée à la configuration compatible la moins robuste, le moteur cognitif cherchant à vérifier ses connaissances avant de déclencher le processus d'expérimentation (exploration). Si C_{14} vérifie la contrainte de QoS, le moteur cognitif identifie les configurations plus robustes afin d'inférer sur la compatibilité de ses configurations parents avec le problème. On a ainsi $TE^m = SC$ et $CME^m = -1$ pour $m = \{1, 2, 4, 5, 10, 11, 13, 14\}$. Si C_{14} n'est pas assez performante, le moteur cognitif identifie les configurations moins robustes afin d'inférer sur l'incompatibilité de ses configurations enfants avec le problème. On a ainsi $TE^m = VC$ et $CME^m = 1$ pour $m = \{14, 15, 17, 18, 23, 24, 26, 27\}$. Une seconde mauvaise décision avec la configuration C_{20} ferait passer $\{TE^m\}_{m=20,21}$ de I à VC sans modifier les autres éléments de **TE**. Les compteurs CME^m serait à 1 pour $m = \{14, 15, 17, 18, 20, 21\}$, à 2 pour $m = \{23, 24, 26, 27\}$ et à 0 pour les autres éléments de **CME**.

Méthodes d'exploration

Le processus d'expérimentation adopte le même comportement défensif que pour un espace de conception totalement ordonné. Il régresse d'un nombre prédéfini A_{def} de rangs lorsqu'il se

met à la recherche d'une configuration plus robuste que la configuration testée. La méthode d'exploration offensive doit par contre être révisée pour tenir compte des incertitudes sur les relations d'ordre prédites. La robustesse croissante des configurations le long de l'échelle de performance n'est plus garantie. Le processus d'expérimentation ne peut donc plus se limiter à une progression rang par rang pour identifier la configuration compatible la moins robuste. Une logique plus complexe est nécessaire comme l'illustre l'exemple de la figure 5.4 s'appuyant sur les notations de la figure 5.2.

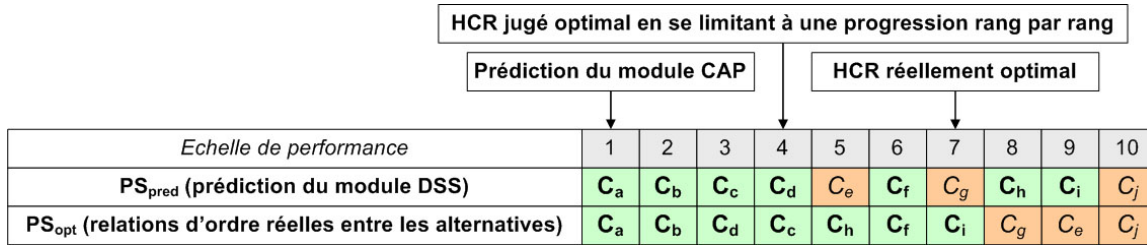


FIGURE 5.4 – Mise en évidence des limites du processus d'expérimentation présenté au chapitre 4 en cas d'erreurs de classement

Le modèle prédictif est supposé vierge de toute connaissance donc le moteur cognitif adopte une attitude prudente en testant la configuration la plus robuste disponible. Il active ensuite le processus d'expérimentation afin de caractériser la région visitée. Avec une progression rang par rang, le processus pense avoir identifié la configuration compatible la moins robuste (*i.e.* C_d) après avoir observé que C_e ne vérifie pas les contraintes. Cette conclusion est clairement sous-optimale car C_f , C_h et C_j vérifient la contrainte de QoS.

Pour repérer ces erreurs de classement, le processus d'expérimentation est autorisé à explorer des rangs éloignés de son estimation actuelle du HCR. L'exploration reste néanmoins contrôlée de manière à minimiser les risques d'expérimentation (*e.g.* C_g et C_j conduisent à une mauvaise décision). Ce principe de précaution, déjà exposé à la section 4.2.3, est nécessaire pour ne pas trop dégrader l'expérience de l'utilisateur lors de sa communication. Le moteur cognitif peut alors compter sur les informations d'exploration et sur le graphe de robustesse embarqué pour construire un plan d'expérience intelligent allant dans ce sens.

Le processus d'expérimentation commence par repérer les opportunités de progression en observant le tableau d'exploration pour les configurations au-delà du HCR actuel (exploration offensive). Il sélectionne alors toutes les configurations associées à l'état I .

Les choix disponibles sont ensuite analysés à l'aide du graphe de robustesse afin d'identifier les configurations ayant la plus grande probabilité de vérifier les contraintes (attitude prudente). Le processus d'expérimentation observe les relations d'ordre entre les alternatives sélectionnées afin de rejeter toutes les configurations dominées (la configuration C_i est dite dominée par C_j s'il existe un lien orienté de C_i vers C_j).

Une configuration dominant toutes les autres est forcément sélectionnée. S'il existe plusieurs configurations non dominées, le processus d'expérimentation fait appel à sa mémoire d'exploration pour déterminer la configuration ayant la plus grande chance de succès. Il favorise la configuration associée au rang le plus faible tant que son poids d'exploration prend la valeur minimale parmi les opportunités considérées. Cette règle déterministe traduit la confiance initiale du moteur cognitif dans les prédictions du module DSS (une configuration de rang plus faible est supposée être plus robuste).³²

32. Nous proposons un exemple d'application de cette règle en annexe B.1.

L'opportunité de plus faible rang peut ne plus s'avérer judicieuse après plusieurs tentatives infructueuses. Le moteur cognitif garde une trace de ces erreurs passées en incrémentant le poids d'exploration de la configuration correspondante. Le processus d'expérimentation peut alors tenter d'explorer d'autres alternatives pour repérer les configurations compatibles isolées et ainsi corriger les erreurs de classement. L'opportunité de plus faible rang ne peut pas non plus être totalement écartée car elle peut s'avérer suffisamment performante pour le nouveau problème considéré. Les résultats d'expérimentation peuvent en effet varier (légèrement) entre deux problèmes proches s'ils se trouvent de part et d'autre d'une frontière de décision.

Ce dilemme est résolu par une prise de décision stochastique basée sur une distribution de probabilité $\{p_i\}_{i=1,\dots,N_{ND}}$ calculée à partir des poids d'exploration $\{w_i\}_{i=1,\dots,N_{ND}}$ associés aux N_{ND} configurations non dominées :

$$p_i = \frac{1}{\sum_{j=1}^{N_{ND}} \frac{w_i - w_{\min} + 1}{w_j - w_{\min} + 1}} \quad (5.2)$$

Conformément à notre interprétation des poids d'exploration, la probabilité p_i de choisir une configuration donnée est d'autant plus faible que son poids w_i est élevé. w_{\min} correspond à la valeur minimale des poids d'exploration considérés. Nous avons décidé en effet de calculer les probabilités avec un poids minimum égal à 1 afin d'accentuer les différences entre les configurations peu explorées de celles largement explorées.³³

Contrôle de l'intégrité de l'échelle de performance

Le processus d'expérimentation mémorise les taux d'erreurs binaires estimés après chaque décision du moteur cognitif afin de comparer les performances des configurations testées et ainsi corriger les erreurs dans les relations d'ordre prédites. L'algorithme développé est décrit à l'encart 3. Il est valable pour tout indicateur de performance à minimiser (*e.g.* BER, latence). Il doit être légèrement modifié (inversion des signes) pour les indicateurs de performance à maximiser (*e.g.* débit). Nous commentons brièvement l'algorithme proposé puis nous proposons deux exemples illustrant les principes appliqués.

Après chaque nouvelle décision, le processus d'expérimentation compare les performances de la configuration testée avec celles des configurations précédemment explorées. L'objectif n'est pas de réaliser un classement absolument exact mais plutôt de produire une échelle fiable pour la prise de décision. Le processus ne compare donc pas les performances des configurations pour des valeurs de BER supérieures à un seuil de comparaison fixé (*e.g.* 10^{-2}). Introduit qualitativement à la section 3.2.3, ce principe s'inscrit dans une logique de tolérance aux imprécisions. Dans le cas présent, il permet de réduire la complexité du système prédictif associé au module DSS en limitant son apprentissage aux relations d'ordre pertinentes pour la prise de décision.

Le processus d'expérimentation modifie les connaissances sur les relations d'ordre lorsqu'une incohérence est repérée dans l'échelle de performance. Deux cas sont envisageables :

- C_i est censée être moins robuste que C_N ($PS_i < PS_N$) mais l'observation des performances relatives indique le contraire ($BER_i > BER_N$). Ce type d'incohérence intervient lors d'une exploration offensive, le processus d'expérimentation cherchant à progresser le long de l'échelle de performance. Nous avons vu que le processus adoptait une attitude prudente dans sa progression afin de limiter les risques d'expérimentation. Les configurations concernées par l'incohérence se limitent donc aux deux configurations comparées.

33. Cette remarque est justifiée à l'aide d'un exemple présentée en annexe B.1.

Algorithme 3 Contrôle de l'intégrité de l'échelle de performance

```

 $N \leftarrow$  Nombre de décisions associées à l'expérience de conception en cours
 $D_N \leftarrow$  Configuration venant d'être testée (Decision)
 $D \leftarrow$  Mémoire stockant les  $N - 1$  précédentes tentatives du moteur cognitif classées par ordre
décroissant de robustesse
 $PS_i \leftarrow$  Rang associé à  $D_i$  dans l'échelle de performance ( $i = 1, \dots, N$ )
 $C_i \leftarrow$  Configuration associée à  $D_i$  ( $i = 1, \dots, N$ )
 $BER_i \leftarrow$  Taux d'erreurs binaires obtenu avec  $C_i$  ( $i = 1, \dots, N$ )
 $BER_{ct} \leftarrow$  Seuil de comparaison (comparison threshold)

Identifier  $PS_N$ 
for  $i = 1$  à  $N - 1$  do
    Identifier  $PS_i$ 
    if  $BER_i < BER_{ct}$  et  $BER_N < BER_{ct}$  then
        if  $PS_i < PS_N$  et  $BER_i > BER_N$  then
            Déplacer  $C_N$  juste avant  $C_i$  dans l'échelle de performance
        else if  $PS_i > PS_N$  et  $BER_i < BER_N$  then
            Identifier les configurations moins robustes que  $C_N$  parmi les alternatives de rang infé-
            rieur à  $PS_i$ 
            Déplacer  $C_N$  et les configurations identifiées juste après  $C_i$  dans l'échelle de performance
            en s'assurant de maintenir leurs relations d'ordre
        end if
    Mettre à jour  $PS_N$ 
end if
end for
Classer les  $N$  décisions par ordre décroissant de robustesse dans  $D$  (i.e.  $PS_1 < \dots < PS_N$ )

```

- C_i est censée être plus robuste que C_N ($PS_i < PS_N$) mais l'observation des performances relatives indique le contraire ($BER_i > BER_N$). Ce type d'incohérence intervient lors d'une exploration défensive, le processus d'expérimentation régressant le long de l'échelle de performance. Une telle incohérence peut aussi s'observer lors de l'exploitation des connaissances si la configuration choisie (optimale d'après l'état de connaissance) n'est pas suffisamment performante en réalité. Le processus doit cette fois utiliser ses capacités d'inférence pour maintenir les relations d'ordre décrites par le graphe de robustesse. Certaines configurations moins robustes que C_N peuvent alors se retrouver elles aussi concernées par l'incohérence repérée.

Les deux cas d'incohérence sont illustrés à la figure 5.5 en reprenant le graphe de robustesse de la figure 3.10. La configuration précédemment explorée (C_i) est mise en valeur en gras. Les configurations en italique sont concernées par l'incohérence soit par comparaison directe des performances (C_N) soit par inférence (configurations moins robustes que C_N). Le processus ne dispose pas d'informations particulières sur les configurations en vert donc il maintient l'ordre prédit tant que celui-ci n'est pas mis à défaut. Pour plus de clarté, les relations d'ordre du graphe sont reproduites à l'aide de liens orientés dans la figure.

Type d'incohérence détectée $PS_i < PS_N$ & $BER_i > BER_N$	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">D_i ↓</div> <div style="text-align: center;">D_N ↓</div> </div>					
Echelle de performance avant correction	...	C_{13}	C_{12}	C_7	C_8	C_{20}
	...	10	11	12	13	14
Echelle de performance après correction	...	C_{20}	C_{13}	C_{12}	C_7	C_8

(a)

Type d'incohérence détectée $PS_i > PS_N$ & $BER_i < BER_N$	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">D_N ↓</div> <div style="text-align: center;">D_i ↓</div> </div>					
Echelle de performance avant correction	...	C_{13}	C_{12}	C_{22}	C_{14}	C_{20}
	...	10	11	12	13	14
Echelle de performance après correction	...	C_{12}	C_{20}	C_{13}	C_{22}	C_{14}

(b)

FIGURE 5.5 – Illustration de l'algorithme 3 après la détection d'une incohérence de type (a) ou (b)

Mise à jour de l'échelle de performance après une exploration offensive réussie

L'exploration offensive n'est plus réalisée selon une progression rang par rang donc il peut s'avérer nécessaire de réorganiser l'échelle de performance en attendant d'expérimenter les autres opportunités de progression. La figure 5.6 illustre la méthode adoptée.

	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">HCR_{Avant} ↓</div> <div style="text-align: center;">Rang exploré ↓</div> </div>					
Echelle de performance avant réorganisation	...	C_{13}	C_{12}	C_7	C_8	C_{20}
	...	10	11	12	13	14
Echelle de performance après réorganisation	...	C_{13}	C_{20}	C_{12}	C_7	C_8

$HCR_{Après}$ ↑

FIGURE 5.6 – Exemple de réorganisation de l'échelle de performance après une exploration offensive réussie

La réorganisation peut venir perturber des relations d'ordre correctement prédites (*e.g.* C_{12} peut effectivement être plus robuste que C_{20}) mais le moteur cognitif ne peut en être sûr sans le vérifier par l'expérimentation. Par précaution, il place C_{20} avant C_{12} puisqu'il est sûr que C_{20} fait partie des configurations compatibles avec le service. Le contrôle de l'intégrité de l'échelle de performance pourra révéler éventuellement que C_{20} aurait dû être placé avant C_{13} (voire encore avant).

Le moteur cognitif continue son exploration offensive tant qu'elle donne lieu à de bonnes décisions. Il a donc l'opportunité de tester C_{12} (ou d'autres configurations) afin de mettre à jour les relations d'ordre si elles s'avèrent erronées.

Prise de décision sur l'arrêt ou la poursuite de l'expérimentation

Le processus d'expérimentation doit stopper la procédure d'essai/erreur lorsque son impact devient pénalisant pour la qualité de service perçue par l'utilisateur. La recherche des configurations compatibles est une opération qui comporte des risques surtout si le moteur cognitif dispose d'une certaine expérience sur son espace de conception. Le nombre de configurations compatibles isolées diminue fortement dans ce cas. Le processus d'expérimentation a donc de fortes chances de prendre une mauvaise décision. Les poids d'exploration permettent de limiter en partie cette situation d'erreur en orientant le processus vers les régions les plus prometteuses de l'espace de conception. Mais la nature aléatoire de la méthode d'exploration adoptée ne garantit pas toujours un choix optimal.

Le moteur cognitif recense les mauvaises décisions prises en incrémentant le compteur n_{BD} à chaque fois qu'une configuration testée ne vérifie pas les contraintes. Les chances d'exploration sont réduites au fur et à mesure que n_{BD} augmente. Le moteur cognitif doit en effet retourner à une configuration suffisamment performante pour limiter la dégradation du service délivré. Il favorise donc l'exploitation des connaissances déjà validées au travers d'expérimentations passées.

Le moteur cognitif doit aussi tenir compte de son expérience sur la région visitée. La nécessité d'explorer se quantifie à l'aide de la probabilité locale d'exploration $P_{Exploration}$ estimée par le module CAP. L'expérimentation est donc favorisée si l'espace local ne semble pas suffisamment bien caractérisé par le moteur cognitif ($P_{Exploration}$ est grande).

La poursuite de l'expérimentation est soumise à une prise de décision stochastique intégrant les deux facteurs présentés (*i.e.* limiter la dégradation de QoS, prolonger l'exploration pour les régions mal caractérisées). Le moteur cognitif tire un nombre aléatoire p qu'il compare au seuil d'exploration $s_{Exploration}$ calculé à l'aide de l'équation suivante :

$$s_{Exploration} = P_{Exploration} \times \exp\left(-\frac{n_{BD}}{n_{BD}^{tol}}\right) \quad (5.3)$$

Le moteur cognitif poursuit l'expérimentation si $p > s_{Exploration}$. Il active le processus de décision dans le cas contraire. Le paramètre n_{BD}^{tol} traduit la tolérance du service sélectionné aux mauvaises décisions. Sa valeur est supposée précisée dans le profil de l'application. De manière générale, elle est plus faible pour les services temps-réel que pour les services non temps-réel moins exigeants en termes d'interaction avec l'utilisateur.

5.2.3 Processus d'apprentissage

Le processus d'apprentissage met à jour les connaissances entretenues par le module DSS (échelle de performance, mémoire d'exploration) en plus de celles maintenues par le module CAP (configuration compatible la moins robuste, probabilité locale d'exploration).

Mise à jour des systèmes connexionnistes évolutifs

Le processus d'expérimentation actualise l'échelle de performance ainsi que le rang de plus haute compatibilité en fonction des retours de l'environnement (*cf.* sec. 5.2.2). Le processus d'apprentissage se sert alors de ces informations pour guider l'évolution des modèles ECF. Il entraîne le système connexionniste du module DSS à l'aide de l'échelle observée en fin d'expérience de conception PS_{final} après l'avoir comparée à l'échelle de performance prédite $PS_{predicted}$. La procédure d'apprentissage est légèrement modifiée car le système connexionniste ne dispose d'aucune classe prédéfinie dans sa dernière couche. Le processus d'apprentissage se charge en effet d'ajouter

une classe supplémentaire à chaque fois que le processus d'expérimentation identifie une nouvelle échelle de performance. Cette action n'était pas nécessaire pour le système connexionniste du module CAP affectant une classe à chacune des configurations disponibles. En dehors de cette étape additionnelle, l'algorithme d'apprentissage reste le même qu'à la section 4.2.4. L'indice de la configuration compatible la moins robuste $(HCR_index)_{final}$ est déduit du rang de plus haute compatibilité final HCR_{final} et de PS_{final} .

Mise à jour de la stratégie locale d'exploration

La mise à jour de la stratégie locale d'exploration a été légèrement revue pour tenir compte des incertitudes sur la structuration de la région de l'espace de conception visité. Les probabilités des automates concernés sont adaptées de la même manière qu'à la section 4.2.4 sauf que la nature du renforcement et son intensité sont désormais déterminées à partir de l'algorithme 4 plutôt que l'algorithme 2. Dans l'algorithme 4, $N_{exploration}$ fait référence au nombre d'explorations tentées par le processus d'expérimentation. $N_{incoherence}$ correspond au nombre d'incohérences repérées entre PS_{final} et $PS_{predicted}$. Ce nombre est calculé en comparant les relations d'ordre entre l'échelle de performance initiale et sa version finale, le compteur étant incrémenté dès qu'une relation a été modifiée. Par exemple, le passage de l'échelle $C_a \succ C_b \succ C_c \succ C_d \succ C_e \succ C_f$ à l'échelle $C_a \succ C_d \succ C_c \succ C_e \succ C_b \succ C_f$ conduit à $N_{Inc} = 4$ car l'expérimentation n'a pas préservé les relations entre C_b et C_c , entre C_b et C_d , entre C_b et C_e ainsi qu'entre C_c et C_d .

Algorithme 4 Détermination des informations pour la mise à jour de la stratégie stochastique d'exploration dans un espace de conception partiellement ordonné

```

if  $(HCR\_index)_{final} = (HCR\_index)_{predicted}$  et  $N_{incoherence} = 0$  then
  L'exploitation est renforcée
  L'intensité de renforcement est égale à  $\max(1, N_{exploration})$ 
else
  L'exploration est renforcée
  if  $HCR_{final} > HCR_{predicted}$  then
    L'intensité de renforcement est égale à  $\max(HCR_{final} - HCR_{predicted}, N_{Inc})$ 
  else
    L'intensité de renforcement est égale à  $\max(1, N_{incoherence})$ 
  end if
end if

```

Mise à jour des mémoires d'exploration

Le processus d'apprentissage conserve une trace des configurations testées grâce aux mémoires d'exploration associées aux règles du système connexionniste attaché au module DSS. Le processus d'apprentissage commence par mettre à jour la mémoire d'exploration locale **ME** en lui ajoutant le vecteur de correction **CME** entretenu par le moteur cognitif lors de ses expérimentations (addition algébrique). Après correction, il ajuste les poids d'exploration trop faible (*i.e.* inférieurs à 1) pour toujours maintenir la valeur minimum à 1. La mémoire d'exploration révisée sert de mémoire initiale si l'algorithme d'apprentissage décide de créer une règle supplémentaire pour représenter une nouvelle association entre un état du canal radio et son échelle de performance correspondante.

Le processus d'apprentissage actualise ensuite les mémoires d'exploration associées aux règles voisines du problème de conception. Il applique la même procédure que précédemment sauf qu'il

pondère le vecteur de correction par un poids w_k calculé en fonction du rayon de voisinage R_V et de la distance d_k entre la règle considérée Cl_k (cluster k) et le problème traité :

$$\mathbf{ME}_k = \mathbf{ME}_k + \mathbf{CME} \times w_k = \mathbf{ME}_k + \mathbf{CME} \times \exp\left(-\frac{d_k}{R_V}\right) \quad (5.4)$$

Le processus d'apprentissage maintient aussi la valeur minimum des poids d'exploration à 1.

Mécanismes de désapprentissage

La dernière modification du processus d'apprentissage est liée à la fiabilité des informations fournies par le processus d'expérimentation *i.e.* PS_{final} et $(HCR_Idx)_{final}$. Ces informations peuvent en effet être sous-optimales ou erronées comme nous l'avons déjà indiqué à la section 5.1. Nous avons donc mis en place une heuristique supplémentaire pour permettre aux systèmes connexionnistes d'oublier les associations ne s'avérant pas performantes pour la prise de décision. L'heuristique reste simple puisqu'elle se base sur trois informations statistiques mémorisées au niveau de chacune des règles développées par les deux systèmes connexionnistes (modules DSS et CAP). Chaque règle est ainsi caractérisée par son âge A (nombre d'apprentissage réalisés depuis la création du nœud), son nombre d'activation N_{act} (nombre de fois où la règle a été utilisée par le processus d'inférence) et son nombre d'erreurs N_{err} (nombre de prédictions contredites par l'expérimentation). Le processus d'apprentissage met à jour ses trois informations à chaque fois qu'il est déclenché. L'âge est actualisé pour toutes les règles de la base de connaissance alors que l'incrément du compteur d'activations et du compteur d'erreurs ne concerne que les règles utilisées pour l'inférence. Le compteur d'erreurs n'est bien évidemment incrémenté qu'en cas de mauvaise prédiction.

La règle de désapprentissage proposée est décrite par l'algorithme 5. Elle est mise en place au niveau de l'étape 4 de l'algorithme d'apprentissage présenté à la section 4.2.4. Elle est appliquée juste après la réduction du champ d'influence des règles activées lorsqu'elles sont associées à une classe différente de la classe identifiée. Dans l'algorithme 5, A_{test} et η_s sont deux paramètres définis en début de simulation. Le paramètre A_{test} permet de ne pas éliminer les nœuds trop rapidement lorsque le moteur cognitif n'a que peu d'expérience sur son espace de conception. Le paramètre η_s indique la marge d'erreur tolérable avant de prendre la décision d'oublier une règle donnée. Il est optimisé afin d'éviter un encombrement de la base de connaissances avec des règles erronées ou sous-optimales (valeur trop forte) et la perte de connaissances issues d'expériences de conception pertinentes³⁴ (valeur trop faible).

Algorithme 5 Règle de désapprentissage

```

if  $A < A_{test}$  then
  Ne pas éliminer la règle  $Cl_k$  (cluster  $k$ )
else
  Calculer le taux d'échec  $\eta_{err} = \frac{N_{err}}{N_{act}}$ 
  if  $R_k = R_{min}$  et  $\eta_{err} > \eta_s$  then
    Éliminer la règle  $Cl_k$ 
  else
    Ne pas éliminer la règle  $Cl_k$ 
  end if
end if

```

34. Il est possible en effet qu'une règle optimale se retrouve avec un compteur d'erreurs non nul si le cluster a eu besoin de plusieurs essais/erreurs pour trouver la taille optimale de son champ d'influence.

La phase de désapprentissage est importante car elle permet de limiter les risques d'expérimentation en réduisant les associations sous-optimales tout en diminuant le nombre de règles donc la complexité des systèmes connexionnistes.

5.2.4 Algorithme de supervision

L'algorithme de supervision se comporte globalement de la même manière qu'à la section 4.2.5. Le processus de décision est cependant affecté par une nouvelle source d'erreurs liée à la prédiction de l'échelle de performance. L'algorithme doit donc assurer un passage plus intelligent entre les processus de décision et d'expérimentation afin de faire la différence entre une erreur de classement et une prédiction trop confiante du HCR. Dans le premier cas, l'algorithme corrige l'erreur pour mettre à jour ses connaissances (HCR et échelle de performance). Il peut ensuite décider d'explorer l'espace de conception en reconnaissant que la région visitée était mal caractérisée. Il peut aussi préférer exploiter ses connaissances mises à jour en activant une nouvelle fois le processus de décision. Dans le deuxième cas, l'algorithme choisit de régresser le long de l'échelle de performance afin de découvrir une configuration plus robuste compatible avec le service.

Le fonctionnement de l'algorithme révisé est présenté à la figure 5.7.

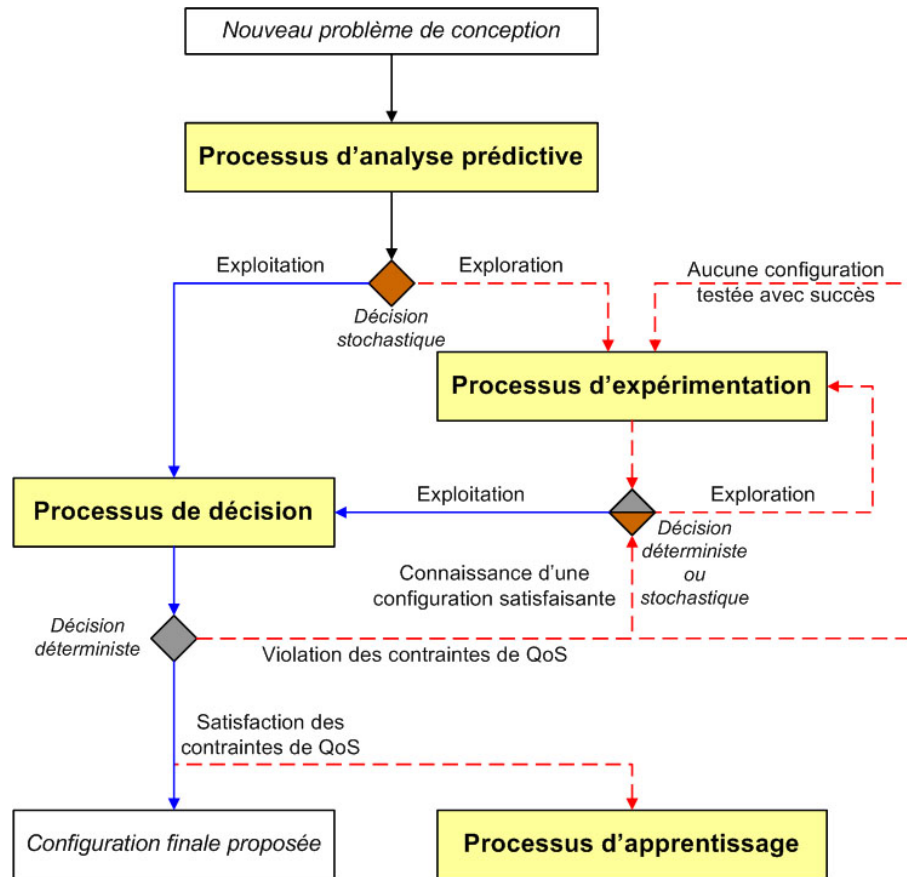


FIGURE 5.7 – Algorithme de supervision pour un espace de conception partiellement ordonné

Le losange en sortie du processus d'expérimentation représente un critère d'arrêt à la fois déterministe et stochastique car nous avons vu que le moteur cognitif pouvait prendre la décision de stopper l'exploration soit par déduction logique (*e.g.* il découvre que le problème n'a

pas de solution parmi les configurations compatibles) soit de manière aléatoire (*cf.* sec. 5.2.2). La flèche supplémentaire entre le processus de décision et ce losange est liée à la démarche passive d'exploration exposée plus haut. Le moteur cognitif est censé avoir déclenché son processus d'expérimentation pour réagir aux retours de l'environnement jusqu'à avoir identifié au moins une configuration compatible avec le service. Le processus de décision est ensuite activé mais la configuration sélectionnée s'avère trop peu performante. Cette erreur d'exploitation n'est pas envisageable avec une échelle de performance fiable. Elle peut intervenir pour les espaces de conception partiellement ordonnés. Le moteur cognitif se retrouve alors à nouveau face au dilemme exploration/exploitation.

De manière plus précise, l'algorithme de supervision applique l'algorithme 6 pour gérer le passage entre le processus de décision et le processus d'expérimentation lorsque la configuration supposée optimale n'atteint pas les exigences de QoS. L'algorithme décrit comment le moteur cognitif met à jour ses connaissances (adaptation du HCR, réorganisation de l'échelle de performance) et il précise les actions à entreprendre pour la suite de l'expérience de conception.

Algorithme 6 Règles de passage entre le processus de décision et le processus d'expérimentation après une exploitation erronée des connaissances

```

 $PS_{exploré} \leftarrow$  Rang de la configuration testée (supposé optimale par le processus de décision)
 $HCR \leftarrow$  Croyance actuelle du moteur cognitif sur le rang de plus haute compatibilité
if  $PS_{exploré} = HCR$  then
    Activer le comportement défensif du processus d'expérimentation
else if  $PS_{exploré} < HCR$  then
    Observer le tableau d'exploration pour les configurations associés aux rangs  $PS$  compris
    entre  $PS_{exploré}$  et  $HCR$ 
    Parmi elles, identifier le nombre de configurations  $N_{SC-I}$  associées à un état différent de
     $VC$ 
    if  $N_{SC-I} = 0$  then
        Adapter le HCR en faisant  $HCR_{new} = PS_{exploré}$ 
        Activer le comportement défensif du processus d'expérimentation
    else
        Réorganiser l'échelle de performance comme illustré à la figure 5.8
        Adapter le HCR en faisant  $HCR_{new} = HCR_{old} - N_{SC-I}$ 
        if Le processus d'expérimentation n'a pas encore été déclenché then
            Tester la configuration associée au nouveau HCR
            Activer le processus d'expérimentation pour réagir aux retours de l'environnement
        else
            Prendre une décision sur le dilemme exploration/exploitation
            if L'exploration est favorisée then
                Activer le comportement offensif du processus d'expérimentation pour rechercher
                d'éventuelles configurations compatibles isolées
            else
                Activer le processus de décision afin de sélectionner la configuration jugée optimale
            end if
        end if
    end if
end if

```

		Rang exploré				HCR _{Avant}	
Echelle de performance avant réorganisation	...	C ₁₃	C ₁₂	C ₂₂	C ₁₄	C ₂₀	...
	...	10	11	12	13	14	...
Echelle de performance après réorganisation	...	C ₁₂	C ₂₀	C ₁₃	C ₂₂	C ₁₄	...
			HCR _{Après}				

FIGURE 5.8 – Exemple de réorganisation de l'échelle de performance après une exploitation erronée des connaissances

5.3 Evaluation du moteur cognitif

Les mécanismes cognitifs sont évalués sur une étude de cas plus complexe impliquant l'introduction d'un modèle de canal à évanouissements sélectifs. Le canal est caractérisé par le profil de puissance associé aux multi-trajets ainsi que par son SNR. Le moteur cognitif commande l'adaptation de modulation et de codage mais il peut aussi activer la fonction d'égalisation si nécessaire. Le scénario de communication est représenté à la figure 5.9.

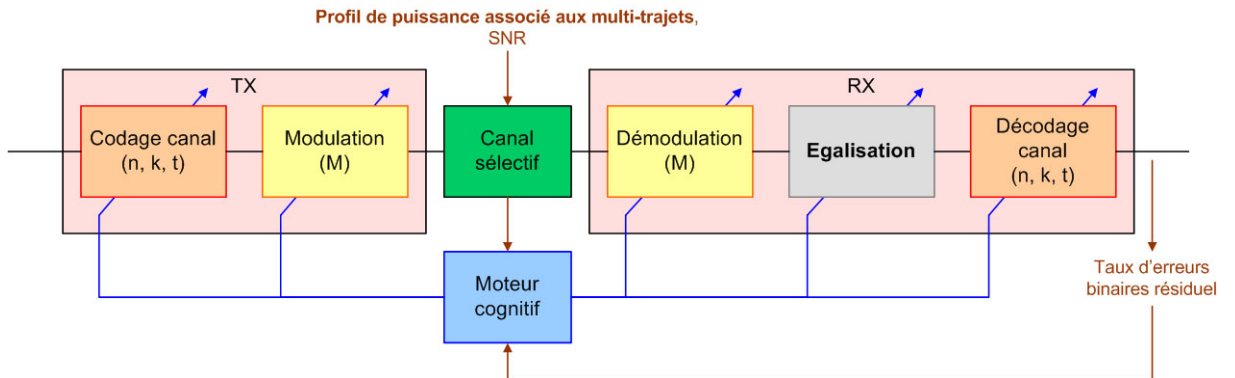


FIGURE 5.9 – Scénario de communication adopté pour l'étude de cas considérée
Les différences par rapport au scénario précédent sont mises en valeur en gras.

Nous avons développé des indicateurs de performance supplémentaires pour analyser l'efficacité du moteur cognitif dans sa structuration de l'espace de conception. Nous présentons ces indicateurs dans un premier temps puis nous commentons les résultats obtenus sur la nouvelle étude de cas.

5.3.1 Présentation d'indicateurs de performance supplémentaires

Nous avons identifié des indicateurs statistiques appropriés pour évaluer les performances du moteur cognitif généralisé. Ces indicateurs sont décrits dans le tableau 5.1.

L'indicateur CG_{12} recense le nombre d'incohérences trouvées à chaque expérience de conception avant de les moyenner sur la fenêtre de calcul. Cet indicateur est lié à la correction des erreurs de classement identifiées.

Les indicateurs CG_{13} à CG_{16} permettent de quantifier les deux sources d'erreurs liées aux démarches active (CG_{13} et CG_{15}) et passive (CG_{14} et CG_{16}) d'exploration. Le nombre d'erreurs

Label	Description des indicateurs de performance	AOS	AOT
Comportement général (CG)			
CG_{12}	Nombre moyen d'incohérences repérées par expérience de conception	Y	Y
CG_{13}	Nombre moyen de configurations compatibles isolées avant expérimentation	Y	Y
CG_{14}	Nombre moyen de configurations jugées compatibles à tort avant expérimentation	Y	Y
CG_{15}	Nombre moyen de configurations compatibles isolées après expérimentation	Y	Y
CG_{16}	Nombre moyen de configurations jugées compatibles à tort après expérimentation	Y	Y
CG_{17}	Proportion de l'expérience de conception dédiée à la prise de décision finale	Y	Y
CG_{18}	Proportion de l'expérience de conception dédiée à la vérification des connaissances	Y	Y
CG_{19}	Proportion de l'expérience de conception passée à s'assurer que le problème n'a pas de solution parmi les configurations disponibles	Y	Y
CG_{20}	Proportion de l'expérience de conception dédiée à la recherche d'une configuration compatible lorsque le modèle prédictif est mis en défaut	Y	Y
CG_{21}	Proportion de l'expérience de conception liée au repérage passif des erreurs de classement	Y	Y
CG_{22}	Proportion de l'expérience de conception liée au repérage actif des erreurs de classement	Y	Y
CG_{23}	Proportion de l'expérience de conception contribuant aux risques d'expérimentation	Y	Y
Fiabilité du modèle prédictif (FMP)			
FMP_4	Fiabilité moyenne des prédictions du module DSS	Y	Y
Complexité du modèle prédictif (CMP)			
CMP_2	Nombre de règles contenues dans la base de connaissances développée par le module DSS	$Y^{(2)}$	$Y^{(2)}$
CMP_3	Nombre total de règles contenues dans les bases de connaissances du moteur cognitif	$Y^{(2)}$	$Y^{(2)}$

TABLE 5.1 – Indicateurs de performance supplémentaires pour l'évaluation du moteur cognitif généralisé

AOS : Calcul de l'indicateur de performance sur l'ensemble de la simulation (Averaged Over Simulation)

AOT : Calcul de l'évolution de l'indicateur de performance au cours de la simulation (Averaged Over Time)

Y/N : Yes/No

(2) Ces indicateurs traduisent déjà une progression donc elles ne sont pas moyennées.

est évalué après l'expérimentation pour CG_{15} et CG_{16} donc leur valeur est toujours inférieure ou égale à celle de CG_{13} et CG_{14} . L'expérimentation est efficace lorsqu'elles prennent des valeurs strictement inférieures.

Les indicateurs CG_7 à CG_{11} sont pertinents pour analyser les réponses de notre moteur cognitif dans le cas d'un espace de conception totalement ordonné. Chacun d'eux représente un comportement particulier que le moteur cognitif active en fonction de son expérience sur la région visitée et des retours de l'environnement. La démarche d'exploration suivie est alors linéaire, le moteur cognitif s'appuyant sur des relations d'ordre fiables pour progresser ou régresser le long de son échelle de performance jusqu'à identifier la solution optimale. La réponse du moteur cognitif peut devenir beaucoup plus complexe pour un espace bénéficiant d'une structuration préalable moins forte. Une analyse plus fine est donc nécessaire afin de mieux comprendre les choix du moteur cognitif. Elle est réalisée au travers des indicateurs CG_{17} à CG_{23} interprétant chacune des décisions formant l'expérience de conception proposée par notre moteur cognitif :

- L'indicateur CG_{17} représente la proportion de l'expérience de conception dédiée à la prise de décision finale. Elle prendra une valeur maximale (*i.e.* 100%) si le moteur cognitif décide de toujours exploiter ses connaissances sans jamais prendre de mauvaise décision.
- L'indicateur CG_{18} comptabilise le nombre de fois où le moteur cognitif a décidé de vérifier ses connaissances actuelles en testant la configuration associée au HCR.
- L'indicateur CG_{19} se focalise sur le nombre de décisions passées à explorer l'espace de conception alors que le problème s'avère impossible à résoudre compte tenu des configurations disponibles.
- L'indicateur CG_{20} regroupe les décisions prises pour revenir à une configuration compatible avec le service lorsque les prédictions conduisent à une mauvaise décision initiale. Elle tient compte de la première configuration choisie ainsi que des réactions défensives déclenchées en réponse à cette mauvaise décision.
- L'indicateur CG_{21} est incrémenté si le processus de décision choisit une configuration trop peu performante alors que le modèle prédictif a été vérifié (ou mis à jour) par l'expérimentation. Cette mauvaise décision résulte d'une erreur de classement repérée ici passivement.
- Les indicateurs CG_{22} et CG_{23} impliquent une démarche active d'exploration, le moteur cognitif faisant des essais pour repérer des configurations compatibles isolées, victimes d'erreurs de classement. Ces essais sont réussis dans le cas de CG_{22} alors qu'ils conduisent à de mauvaises décisions dans le cas de CG_{23} (risques d'expérimentation).

5.3.2 Résultats

Dans ce chapitre, les configurations disponibles résultent de la combinaison complète des options associées aux trois blocs reconfigurables. Elles sont recensées au tableau 5.2. L'environnement radio est simulé à travers un canal à deux trajets modélisé par un filtre statique de réponse impulsionnelle complexe $[1; 0, 4 - 0, 2i]$. Les performances des différentes configurations sont affichées à la figure 5.10 où nous avons mis en valeur l'espace de classification.

Comme pour le chapitre 4, les résultats sont moyennés sur 10 simulations et la progression des indicateurs de performance est calculée sur une fenêtre de longueur $L_W = 1000$. Le moteur cognitif ne contient initialement aucune règle dans sa base de connaissances. Les connaissances expertes sur l'espace de conception sont intégrées dans le graphe de robustesse présenté à la figure 3.10.

Label	Code correcteur	Modulation	Égaliseur	Notation
C_1	BCH(127, 92, 5)	BPSK	MLSE	c1e1m1
C_2	BCH(127, 120, 1)	BPSK	MLSE	c2e1m1
C_3	Aucun	BPSK	MLSE	c3e1m1
C_4	BCH(127, 92, 5)	BPSK	Linéaire	c1e2m1
C_5	BCH(127, 120, 1)	BPSK	Linéaire	c2e2m1
C_6	Aucun	BPSK	Linéaire	c3e2m1
C_7	BCH(127, 92, 5)	BPSK	Aucun	c1e3m1
C_8	BCH(127, 120, 1)	BPSK	Aucun	c2e3m1
C_9	Aucun	BPSK	Aucun	c3e3m1
C_{10}	BCH(127, 92, 5)	QPSK	MLSE	c1e1m2
C_{11}	BCH(127, 120, 1)	QPSK	MLSE	c2e1m2
C_{12}	Aucun	QPSK	MLSE	c3e1m2
C_{13}	BCH(127, 92, 5)	QPSK	Linéaire	c1e2m2
C_{14}	BCH(127, 120, 1)	QPSK	Linéaire	c2e2m2
C_{15}	Aucun	QPSK	Linéaire	c3e2m2
C_{16}	BCH(127, 92, 5)	QPSK	Aucun	c1e3m2
C_{17}	BCH(127, 120, 1)	QPSK	Aucun	c2e3m2
C_{18}	Aucun	QPSK	Aucun	c3e3m2
C_{19}	BCH(127, 92, 5)	8-PSK	MLSE	c1e1m3
C_{20}	BCH(127, 120, 1)	8-PSK	MLSE	c2e1m3
C_{21}	Aucun	8-PSK	MLSE	c3e1m3
C_{22}	BCH(127, 92, 5)	8-PSK	Linéaire	c1e2m3
C_{23}	BCH(127, 120, 1)	8-PSK	Linéaire	c2e2m3
C_{24}	Aucun	8-PSK	Linéaire	c3e2m3
C_{25}	BCH(127, 92, 5)	8-PSK	Aucun	c1e3m3
C_{26}	BCH(127, 120, 1)	8-PSK	Aucun	c2e3m3
C_{27}	Aucun	8-PSK	Aucun	c3e3m3

TABLE 5.2 – Configurations disponibles pour la conception du lien radio

Paramétrage du moteur cognitif

Dans le chapitre 4, nous avons analysé l'influence des paramètres sur le comportement du moteur cognitif. Nous sommes partis de cette analyse pour adapter le paramétrage du moteur cognitif de manière experte. Les valeurs choisies sont résumées au tableau 5.3.

Cette étape de configuration limite l'autonomie décisionnelle du moteur cognitif en nécessitant un effort humain et une bonne connaissance de l'espace de conception. Cette limitation peut cependant être levée en introduisant des mécanismes plus avancés dans le moteur cognitif. Nous proposons à ce titre quelques solutions dans le chapitre 6 (*e.g.* adaptation en ligne des paramètres, fusion des règles redondantes).

L'espace de classification est plus complexe pour cette étude de cas donc nous avons choisi de définir les clusters avec une granularité plus fine ($R_{\min} = 0,005 < 0,1$). De même, nous avons réduit la valeur de R_{\max} pour limiter les généralisations trop brutales en début de simulation. Les frontières de décision sont beaucoup plus nombreuses donc l'espace de classification sera couvert par une densité plus grande de règles. Nous avons donc diminué la valeur de R_V pour

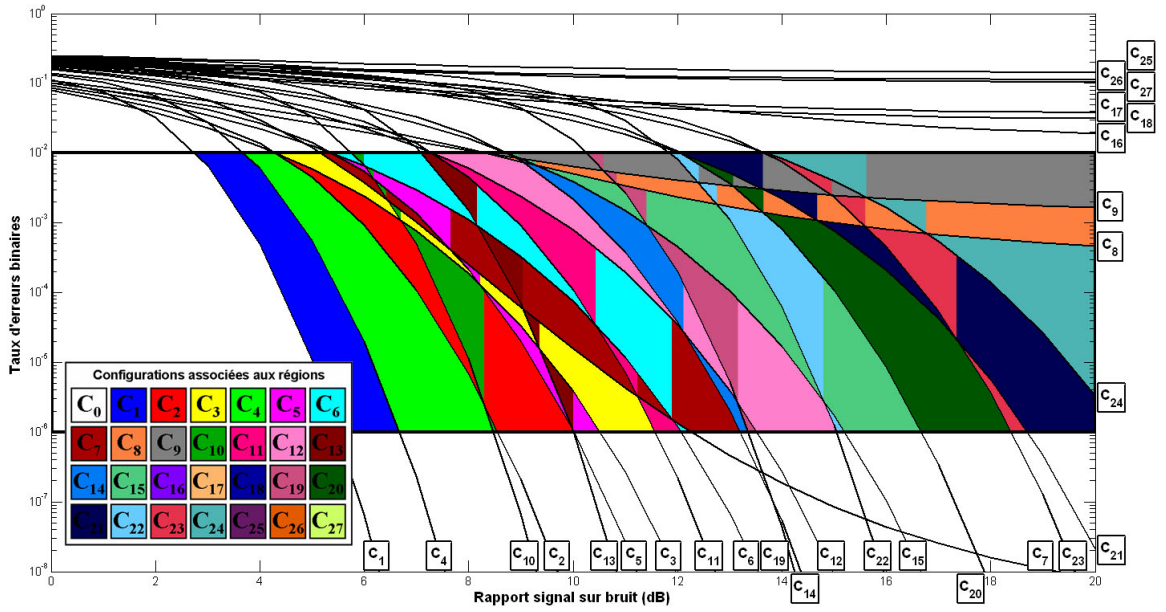


FIGURE 5.10 – Espace de classification associé au nouveau scénario de reconfiguration

RALFE		DSS					CAP						
D	n_{BD}^{tol}	R_{max}	R_{min}	$M-of-N$	R_V	η_s	P_{CC}	R_{max}	R_{min}	$M-of-N$	R_V	a	η_s
1	3	0,4	0,005	3	0,05	0,5	0,1	0,25	0,005	3	0,05	0,15	0,5

TABLE 5.3 – Paramètres du moteur cognitif

limiter le nombre d'automates impliqués dans la stratégie locale d'exploration. En contrepartie, nous avons choisi une valeur plus grande de a afin de réduire les risques d'expérimentation. Le paramètre n_{BD}^{tol} traduit la tolérance du service sélectionné aux mauvaises décisions. Nous avons décidé de prendre une valeur constante relativement grande pour favoriser l'exploration en début de simulation. L'oubli des règles est enfin contrôlée à partir du paramètre η_s . Une règle est éliminée si son rayon prend la valeur minimale et si le taux d'échec dépasse 0,5.

Qualité des décisions prises par notre moteur cognitif

Nous n'avons pas retranscrit la matrice de confusion pour des raisons de place (matrice 28×28). Son allure est similaire à la matrice affichée pour la première étude de cas. En moyenne, notre moteur cognitif prend une décision optimale (en fin d'expérience de conception) pour 19408 problèmes sur les 20000 problèmes traités. Il se trompe sur la compatibilité du problème seulement pour 1,6% des problèmes associés à la configuration la plus robuste (40,1 problèmes sur 2486). Cette situation problématique est donc évitée dans la majorité des cas. Comme au chapitre 4, la valeur de cet indicateur pourrait être minimisée en diminuant P_{CC} .

Cette première analyse valide les extensions proposées dans ce chapitre. Les figures 5.11 à 5.22 fournissent une analyse détaillée du comportement de notre moteur cognitif dans un espace de conception partiellement ordonné.

Le moteur cognitif évolue dans un espace plus complexe caractérisé par des régions de classification plus nombreuses et plus petites. Il a donc besoin d'un plus grand nombre d'expériences

pour apprendre à le caractériser avec précision. Ses explorations sont plus fréquentes ce qui augmente les chances de mauvaises décisions comme en atteste la figure 4.5. La courbe évolue donc moins rapidement vers un pourcentage élevé par rapport à l'étude de cas précédente.³⁵ En fin de simulation, 80,9% des problèmes sont résolus sans prendre aucune mauvaise décision (contre 94,8% auparavant). La pente est cependant plus élevée donc le moteur cognitif n'a pas fini de s'améliorer avec l'expérience.³⁶

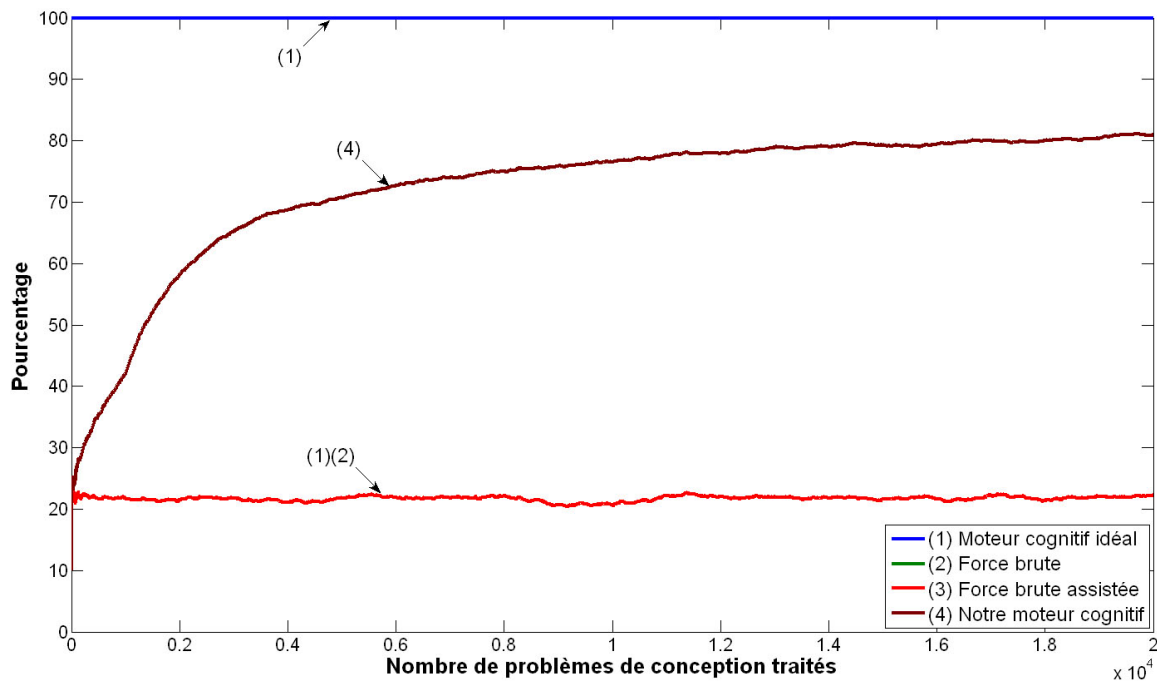


FIGURE 5.11 – Pourcentage d'expériences de conception n'impliquant aucune mauvaise décision lors de la phase de recherche d'une configuration adaptée (CG_2)

Les figures 5.12 et 5.13 montrent que notre moteur cognitif est proche du comportement optimal malgré la plus grande complexité de l'espace de conception. Sur l'ensemble de la simulation, le moteur cognitif termine ses expériences de conception par des décisions proches du rang optimal dans l'échelle d'optimalité ($CG_4 = 0,11$ en moyenne). Sur ces décisions finales, 97,2% sont même optimales. Ces résultats confirment la pertinence des extensions proposées donc la généralité des mécanismes cognitifs développés lors de cette thèse.

35. La comparaison avec l'étude de cas précédente n'a pas lieu d'être *a priori* car elle fait intervenir à la fois un changement du moteur cognitif, de ses paramètres et de l'espace de conception. Elle permet néanmoins d'illustrer les baisses de performance légitimes liées à la complexité additionnelle de l'espace de conception et au besoin de le structurer par apprentissage.

36. Nous avons stoppé la simulation après 20000 problèmes alors que le moteur cognitif a encore une marge de progression importante. Cette décision résulte de l'esprit dans lequel nous évaluons le moteur cognitif. A travers cette thèse, nous cherchons à introduire de nouveaux concepts pour la prise de décision autonome par une radio. Les mécanismes cognitifs simulés ne sont pas optimisés même si nous avons identifié de nombreuses perspectives pour améliorer leurs performances (*cf.* chapitre 6). Dans le cadre de la thèse, nous préférons raisonner en termes de preuve de concept, notamment par rapport aux capacités d'apprentissage et à l'autonomie décisionnelle du moteur cognitif.

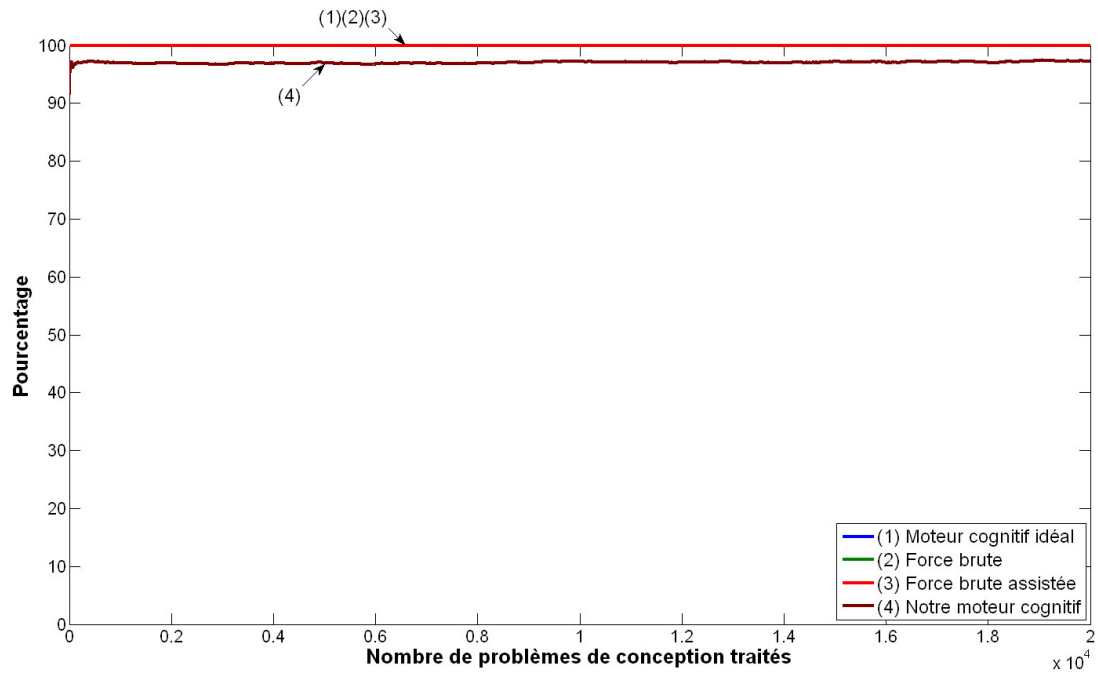


FIGURE 5.12 – Pourcentage d’expériences de conception se terminant par une décision optimale (CG_3)

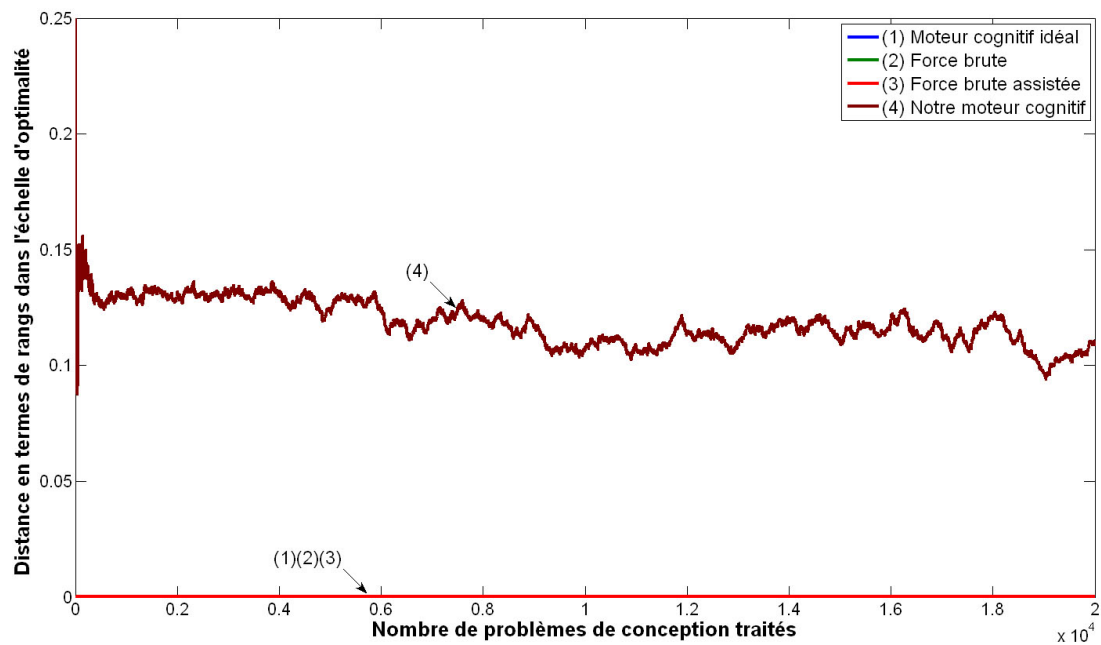


FIGURE 5.13 – Distance moyennée au comportement optimal en termes de rangs dans l’échelle d’optimalité (CG_4)

Efficacité du moteur cognitif dans sa recherche d'une configuration adaptée

La figure 5.14 affiche l'évolution du nombre de décisions prises par expérience de conception. La courbe de progression du moteur cognitif garde la même allure que pour l'étude de cas précédente mais elle décroît plus lentement en raison de la complexité supplémentaire de l'espace de conception. Le moteur cognitif a tendance en effet à recourir à l'expérimentation plus souvent ce qui se traduit par un nombre moyen de décision plus grand (1,51 décisions en fin de simulation contre 1,16 auparavant).

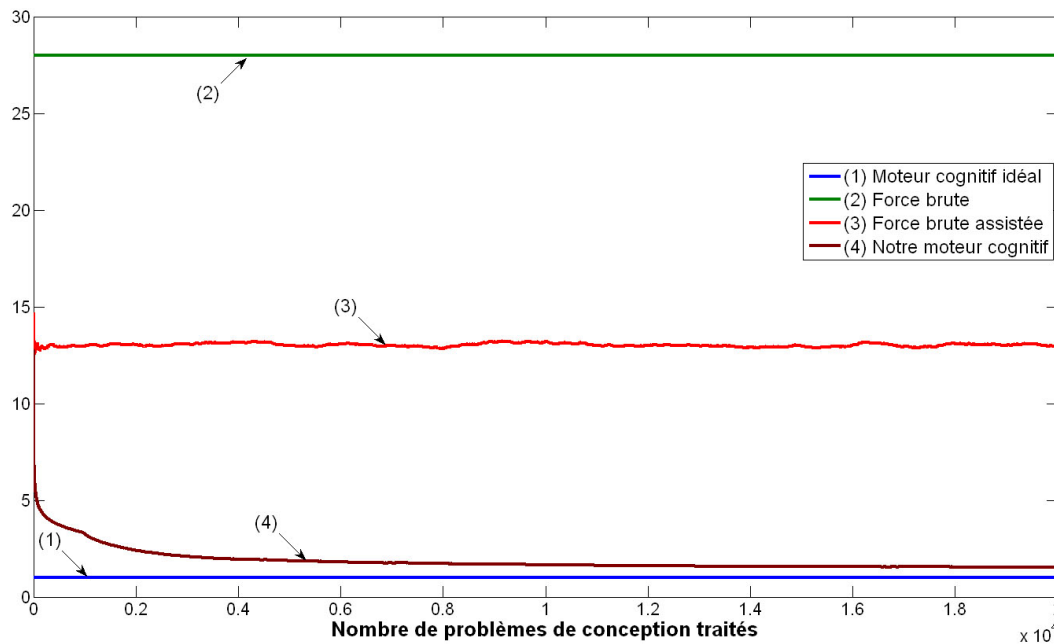


FIGURE 5.14 – Nombre moyenné de décisions prises par expérience de conception (CG_5)

Cette conclusion est aussi valable pour la figure 5.15 se focalisant sur les mauvaises décisions. En fin de simulation, le nombre moyen de mauvaises décisions passe ainsi de 0,052 à 0,262 entre les deux études de cas. Cette valeur reste satisfaisante. De plus, elle peut être rendue beaucoup plus faible par optimisation des mécanismes cognitifs. Nous proposons ici deux solutions complémentaires déjà évoquées au chapitre précédent :

- Dans son implémentation actuelle, le processus d'expérimentation prolonge automatiquement son exploration après avoir testé une configuration vérifiant la contrainte de performances (*cf.* fig. 5.3). Ce comportement est profitable en début de simulation car il permet au moteur cognitif de caractériser son espace de conception plus rapidement. Il devient défavorable après un certain temps lorsque le moteur cognitif se contente d'affiner son modèle prédictif. Sa marge de progression est beaucoup plus faible donc il serait préférable de minimiser les risques d'expérimentation en contrôlant le dilemme exploration/exploitation après chaque décision au lieu d'attendre qu'une mauvaise décision ne soit prise.
- La mise à jour de la stratégie stochastique d'exploration peut être optimisée par une adaptation en ligne des paramètres associés à l'équipe d'automates d'apprentissage. Le paramètre de renforcement peut être ajusté localement (*i.e.* pour chaque automate) à partir d'informations statistiques sur les succès et les échecs d'exploration. Le ratio échec/succès augmente progressivement avec l'expérience du moteur cognitif car le nombre de configurations compatibles isolés diminue progressivement. Le processus d'apprentissage peut donc

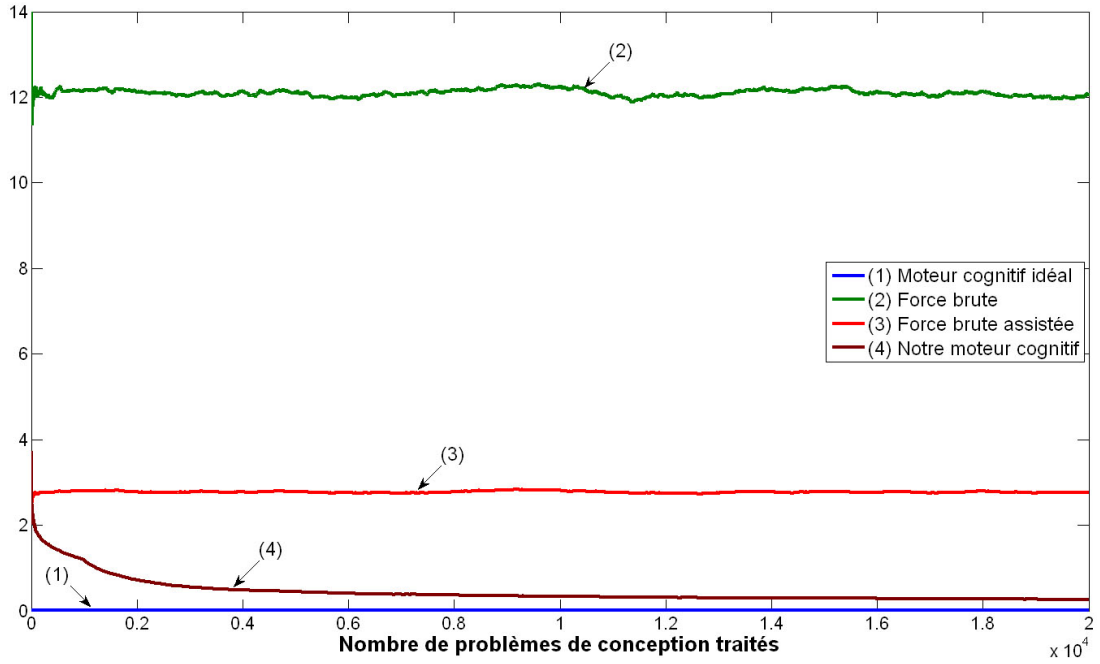


FIGURE 5.15 – Nombre moyenné de mauvaises décisions prises par expérience de conception (CG_6)

exploiter cette indication qualitative pour augmenter la valeur de a afin de diminuer plus rapidement la probabilité d'exploration et ainsi limiter les risques d'expérimentation. De la même manière, il serait intéressant de réduire le rayon de voisinage après un certain temps afin de maintenir la stabilité des régions bien caractérisées.

L'apport des connaissances expertes est encore plus flagrant avec un espace de conception complexe. L'exploration par force brute gagne considérablement en efficacité suite à l'introduction du graphe de robustesse. Le nombre de décisions passe ainsi de 28 à 12,9 décisions pour un gain de 53,9% et le nombre de mauvaises décisions est réduit encore plus en passant de 12,06 à 2,67 (gain de 77,9%). Une structuration plus forte ne ferait qu'améliorer ces chiffres comme nous le verrons un peu plus loin.

La figure 5.16 affiche l'évolution du nombre moyen d'incohérences identifiées par expérience de conception. Le processus d'expérimentation met en évidence de nombreuses erreurs de classement en début de simulation. Le moteur cognitif intègre ces informations progressivement afin d'améliorer ses connaissances sur la structuration de l'espace de conception. Les phases d'exploration se font plus rares avec le temps et les échelles prédites se montrent de plus en plus fiables donc le nombre d'incohérences identifiées diminue fortement pour atteindre une valeur de 0,044 en fin de simulation (*i.e.* 44 incohérences ont été repérées sur les 1000 derniers problèmes traités). Ce nombre très faible montre que notre moteur cognitif peut s'affranchir de relations d'ordre totales et se reposer sur ses capacités d'apprentissage pour ranger les alternatives en fonction de leur robustesse vis-à-vis de la contrainte.

La figure 5.17 complète l'analyse sur l'efficacité du processus d'expérimentation dans sa recherche d'un modèle prédictif fiable. Le nombre d'erreurs de prédiction initialement élevé diminue fortement grâce aux phases d'exploration permettant au moteur cognitif d'emmagasiner des connaissances sur son espace de conception. L'efficacité du processus d'expérimentation se mesure de deux manières :

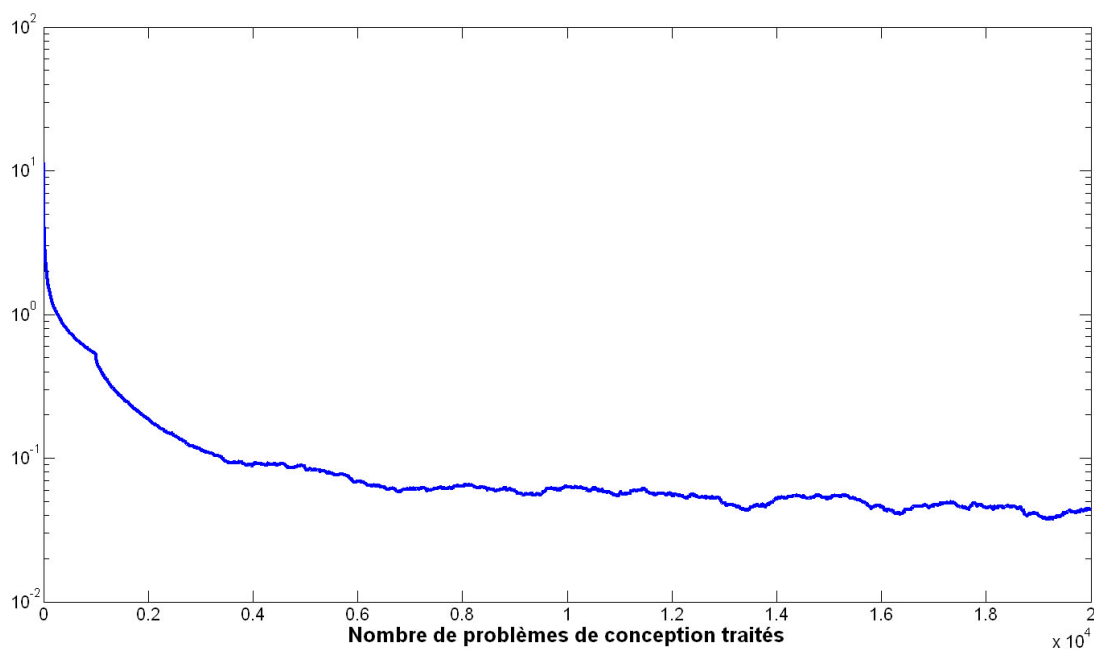


FIGURE 5.16 – Nombre moyenné d'incohérences identifiées par expérience de conception (CG_{12})

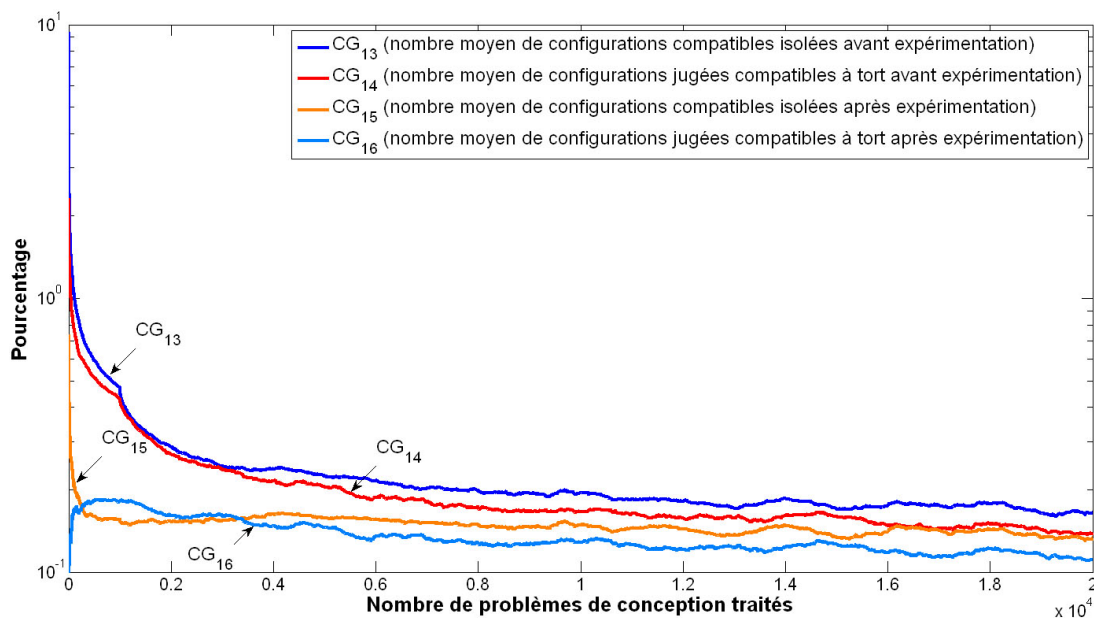


FIGURE 5.17 – Évaluation de l'efficacité du processus d'expérimentation dans sa recherche d'un modèle prédictif fiable (CG_{13-16})

- La décroissance des courbes associées à CG_{13} et CG_{14} montre que les prédictions sont de plus en plus fiables avec l'expérience du moteur cognitif. Le nombre de configurations compatibles isolées passe ainsi de 9,4 en début de simulation à 0,165 en fin de simulation (*i.e.* les 1000 dernières prédictions excluent 165 configurations à tort lors de l'identification des ensembles de configurations compatibles). Le nombre de configurations incompatibles jugées suffisamment performantes passe de 2,33 en début de simulation à 0,138 en fin de simulation (*i.e.* les 1000 dernières prédictions incluent 138 configurations à tort lors de l'identification des ensembles de configurations compatibles). Le modèle prédictif gagne donc considérablement en fiabilité grâce aux actions couplées des processus d'expérimentation et d'apprentissage.
- L'écart entre les courbes avant et après expérimentation quantifie l'apport de l'exploration au fur et à mesure de la simulation. L'écart initialement grand se réduit progressivement conformément aux interprétations précédentes. Sur les 1000 derniers problèmes, le moteur cognitif a repéré 27,4 erreurs de prédiction passivement ($CG_{16} - CG_{14}$) et 32,2 erreurs de prédiction par une démarche active ($CG_{15} - CG_{13}$). Bien que faible, ces nombres montrent que le moteur cognitif a encore besoin d'expérience pour identifier parfaitement les frontières de décision de cet espace complexe.

Analyse des expériences de conception menées par notre moteur cognitif

La figure 5.18 affiche l'évolution des indicateurs CG_{17} à CG_{23} détaillant la nature des décisions prises par notre moteur cognitif lors de ses expériences de conception. La distribution initiale des indicateurs traduit une exploration intensive de l'espace de conception. CG_{17} prend une valeur faible donc le moteur cognitif a besoin d'un grand nombre de décisions avant de se satisfaire d'une réponse acceptable (voire optimale). Son modèle prédictif n'est pas toujours fiable donc le moteur cognitif passe une partie de son expérience à réagir pour rechercher une configuration compatible (*cf.* CG_{20}). Il cherche surtout à se rapprocher du comportement optimal par une démarche active d'exploration. La proportion d'explorations offensives réussies décroît avec le nombre de configurations compatibles isolées (*cf.* CG_{22}). Le moteur cognitif apprend alors à limiter les risques d'expérimentation en ajustant sa stratégie stochastique d'exploration. Il déclenche le processus d'expérimentation moins souvent et il limite plus rapidement ses phases d'exploration en cas de mauvaises décisions (CG_{23} diminue et CG_{17} augmente).

L'indicateur CG_{21} prend toujours une valeur faible pour deux raisons. La découverte passive d'une erreur de classement s'accompagne toujours d'une phase d'exploration pour réagir à la mauvaise décision prise et tenter de mieux caractériser la région de l'espace visitée. Les expériences de conception concernées impliquent donc un grand nombre de décisions ce qui diminue la proportion de l'expérience liée à l'identification de l'erreur de classement. Cette raison est prépondérante en début de simulation lorsque la probabilité d'exploration est encore grande et que l'exploration est souvent couronnée de succès. La découverte passive d'erreurs de classement se fait ensuite plus rare comme en atteste l'évolution de CG_{14} à la figure 5.17. En fin de simulation, le moteur cognitif continue à avoir recours à l'expérimentation ($CG_{17} \neq 100\%$) mais son modèle prédictif est beaucoup plus fiable (CG_{18} est plus grand que CG_{20}) et proche de l'optimal (CG_{22} est très faible par rapport à CG_{23}).

L'évolution de CG_{19} montre enfin que notre moteur cognitif apprend à prendre confiance dans ses prédictions sur l'incompatibilité du problème. La valeur de CG_{19} diminue ainsi progressivement alors que le nombre de problèmes associés à C_0 reste sensiblement constant sur l'ensemble de la simulation (les problèmes sont générés aléatoirement).

Cette figure permet aussi d'avoir une indication sur l'impact des décisions prises par le moteur

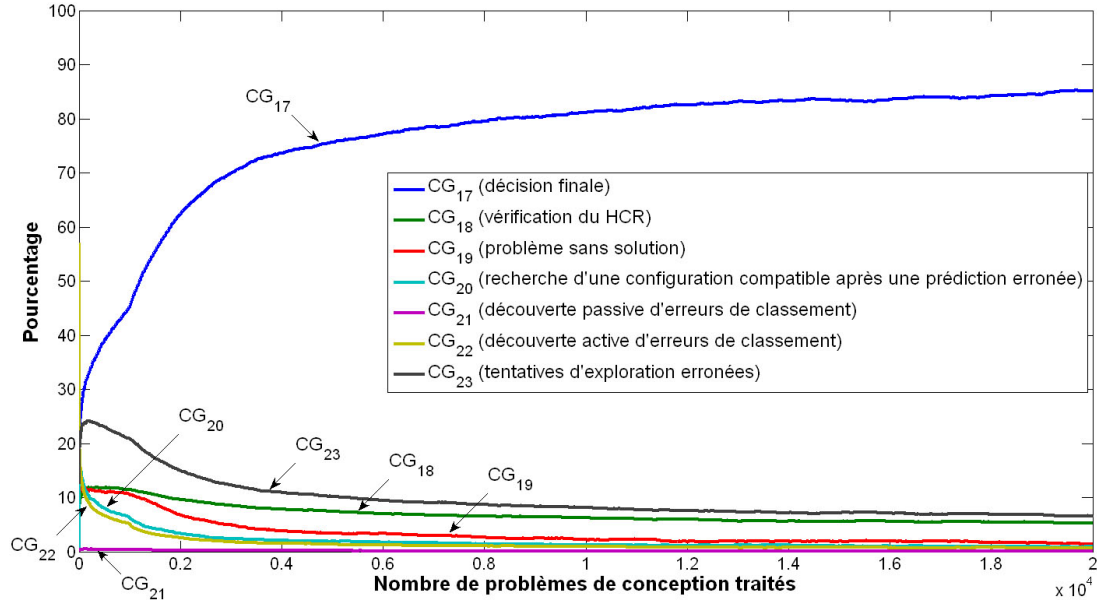


FIGURE 5.18 – Analyse de la réponse du moteur cognitif aux problèmes de conception posés (CG_{17–23})

cognitif. Nous reprenons les notations O (décision optimale), G (décision correcte) et B (mauvaise décision) pour préciser les évaluations associées à chacun des indicateurs. CG_{17} est associée à l'évaluation G ou O en fonction de l'optimalité de la décision finale (souvent O d'après la figure 5.12). De même, CG_{18} correspond à une décision correcte (G) voire optimale (O) si la configuration compatible la moins robuste s'avère être l'alternative la plus satisfaisante. CG_{19} est lié au motif d'évaluation G^n ($n \geq 1$) car les configurations explorées n'ont aucune chance de vérifier les exigences de QoS. Avec CG_{20} , le moteur cognitif prend une mauvaise décision initiale donc il régresse le long de son échelle de performance jusqu'à retrouver une configuration compatible avec la contrainte de performance (motifs d'évaluation B^nG ou B^nO avec $n \geq 1$). CG_{21} donne lieu à une mauvaise décision (B) profitant à la mise à jour du modèle prédictif. CG_{22} a un effet doublement positif puisqu'il permet au moteur cognitif de se rapprocher du comportement optimal sans dégrader la qualité de service (évaluation G ou O). Par contre, CG_{23} a un impact négatif sur l'expérience de l'utilisateur car les configurations sélectionnées ne sont pas suffisamment performantes pour délivrer le service correctement (évaluation B). Nous voyons ainsi qu'en fin de simulation, les mauvaises décisions sont principalement dues aux échecs d'exploration donc il serait préférable de réduire les risques pris en diminuant plus rapidement la probabilité d'exploration.

Evolution de l'expérience du moteur cognitif sur son espace de conception

La figure 5.19 montre que la probabilité moyenne d'exploration diminue beaucoup plus lentement comparé à l'étude de cas précédente. Cette observation est cohérente au regard de la complexité de l'espace de conception. Les frontières de décision sont plus nombreuses et les régions de classification plus petites donc le moteur cognitif a besoin de plus d'expériences pour caractériser son espace. Son modèle prédictif est mis en défaut plus régulièrement (*cf.* CG_{20} et CG_{21}) et ses tentatives d'exploration connaissent plus souvent le succès (*cf.* CG_{22}). Ces deux raisons expliquent la décroissance plus lente de la courbe. En fin de simulation, la probabilité

moyenne d'exploration garde encore une valeur relativement élevée (25,8%) ce qui explique le nombre de mauvaises décisions liées à une expérimentation excessive (*cf.* CG_{23}).

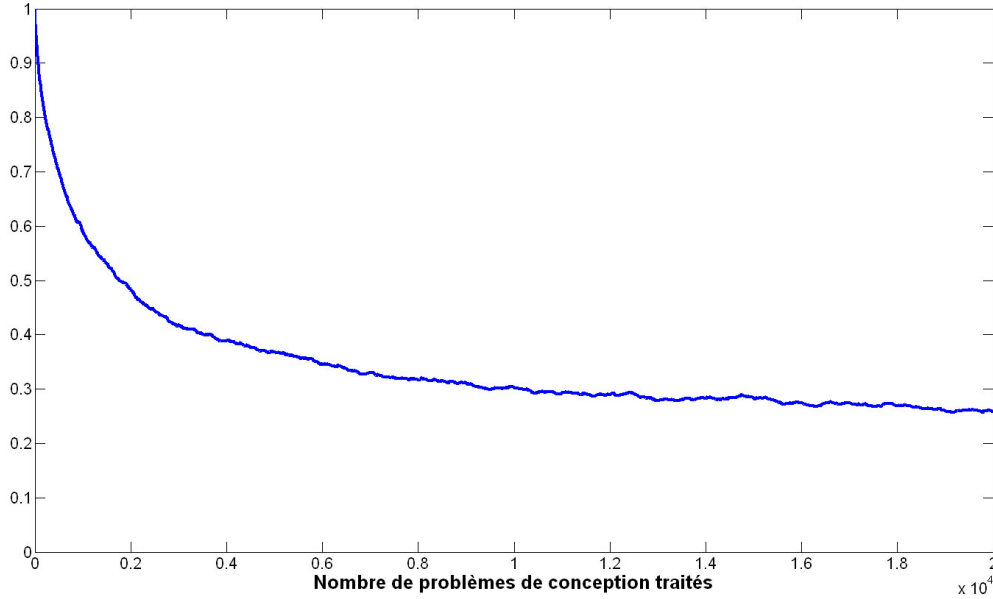


FIGURE 5.19 – Taux d'exploration moyenné sur l'ensemble des automates d'apprentissage ($PSSE_1$)

Les automates d'apprentissage sont mis à jour en considérant une relation de voisinage basée sur la proximité des clusters. La stabilité exploratoire d'une région dépend donc de plusieurs facteurs :

- Le rayon de voisinage définit la portée maximale de la relation de voisinage. La mise à jour des automates est donc plus fréquente avec un large rayon. Ce paramètre intervient aussi dans le calcul du terme de récompense/pénalisation. D'après l'expression 4.11, la probabilité d'exploration est modulée avec une intensité moins forte pour un rayon de voisinage plus faible.
- La position du cluster a une influence sur le nombre d'automates impliqués dans la mise à jour de la stratégie d'exploration. Nous avons adopté une couverture circulaire pour définir la notion de voisinage. Les automates en périphérie de l'espace de classification se retrouvent donc avec un nombre plus faible de voisins par rapport aux automates situés au centre.
- Les règles de mise à jour sont légèrement différentes lorsque le problème s'avère impossible à résoudre. Nous avons traité ce cas avec précaution pour éviter une rupture injustifiée du service (C_0) alors que C_1 peut convenir. Nous noterons d'ailleurs que la probabilité d'exploration diminue moins rapidement pour les faibles valeurs de SNR (*cf.* évolution de la carte d'exploration en annexe B.2).
- La densité des clusters joue à la fois sur le nombre d'automates mis à jour et sur l'intensité du renforcement (l'équation 4.11 fait intervenir la distance du cluster au problème traité). Elle donne aussi une indication sur la difficulté de la région visitée. Le module CAP nécessite en effet un plus grand nombre de règles pour identifier les nombreuses frontières de décision d'une région complexe.
- La nature du renforcement dépend de la pertinence des règles. L'exploitation est favorisée de façon plus soutenue si le modèle prédictif s'avère fiable pour la région considérée.

Fiabilité du modèle prédictif

La figure 5.20 affiche l'évolution de la fiabilité du modèle prédictif et son impact pour la prise de décision. La progression du module CAP est plus lente que pour l'étude de cas précédente car le moteur cognitif évolue dans un espace de conception plus complexe en disposant de moins d'informations sur sa structuration. La précision de ses prédictions est égale à 57,5% pour les 1000 premiers problèmes et de 82,7% pour les 1000 derniers (nous avons obtenu respectivement 82,3% et 92,8% pour l'étude de cas précédente). La tendance finale de la courbe montre néanmoins que le module CAP n'a pas fini de progresser et qu'il peut encore affiner son modèle avec de nouvelles expériences.

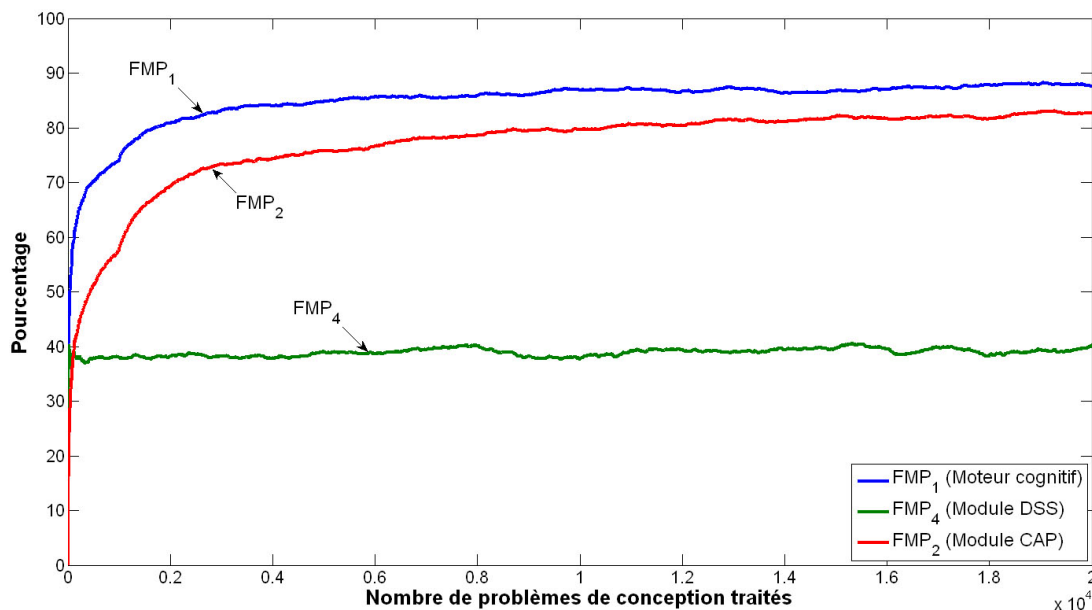


FIGURE 5.20 – Fiabilité des prédictions des modules DSS (FMP_4) et CAP (FMP_2) comparée à la qualité des décisions prises par le moteur cognitif (FMP_1)

Le module DSS n'a pas la même précision dans ses prédictions. La qualité de ses prédictions est évaluée en comparant les relations d'ordre prédites avec l'échelle de performance optimale pour toutes les configurations de rang inférieur ou égal au HCR. Cette méthode d'évaluation met en valeur la tolérance de notre moteur cognitif aux imprécisions. Le module DSS ne cherche pas à identifier toutes les relations d'ordre avec exactitude. Il se contente d'apprendre à structurer localement son espace de conception. Par exemple, il est inutile de connaître les relations d'ordre entre C_2 , C_4 et C_{10} pour des SNR élevés car ces trois configurations sont assurées de vérifier la contrainte pour un canal aussi facile. La fiabilité des relations d'ordre est par contre plus importante pour les configurations de rang plus élevé.

Les erreurs de classement sont mises en valeur en comparant les performances relatives des configurations testées. Le moteur cognitif s'appuie aussi sur son graphe de robustesse pour inférer sur la capacité d'une configuration à délivrer le service correctement. Cette indication qualitative ne permet pas de corriger d'éventuelles erreurs de classement pour les rangs faibles. Elle a par contre l'avantage de grandement accélérer l'exploration de l'espace de conception. Notre moteur cognitif s'inscrit donc totalement dans la philosophie du soft computing prônant la tolérance aux imprécisions pour la mise au point de solutions plus efficaces et plus robustes (*cf.* sec. 3.5).

La tolérance aux imprécisions apparaît de façon plus nette au travers de la courbe associée

au moteur cognitif. Pour les 1000 premiers problèmes, le gain relatif est égal à 38,7%, le taux d'erreurs passant de 42,5% pour le module CAP à 32,7% pour le moteur cognitif. Le gain relatif est plus faible en fin de simulation mais il reste significatif puisqu'il prend une valeur de 28,4% (le taux d'erreurs passe de 17,3% à 12,4% entre les deux courbes).

Comme pour le chapitre précédent, nous affichons la carte d'adéquation à différentes étapes de la simulation afin de montrer l'évolution de la pertinence des règles avec l'expérience du moteur cognitif (figure 5.21).

Évaluation qualitative de la complexité du moteur cognitif

La figure 5.22 montre l'évolution du nombre de règles à l'intérieur des bases de connaissance développées par les modules DSS et CAP. Le nombre total de règles est précisé au travers de la dernière courbe associée au moteur cognitif.

Le moteur cognitif a besoin d'un plus grand nombre de règles qu'auparavant pour caractériser son espace de conception. Cette tendance est logique car il doit apprendre à structurer son espace de conception en plus d'identifier la configuration compatible la moins robuste. De plus, les régions de classification sont plus petites et plus nombreuses donc le module CAP doit couvrir son espace avec un nombre plus important de clusters afin de pouvoir identifier les frontières de décision avec précision. L'espace du module DSS est plus simple à couvrir en comparaison puisque le module ne travaille que sur une seule dimension (il ne prend que la valeur du SNR en paramètre d'entrée). Le nombre de règles évolue donc plus lentement.

Après 20000 problèmes de conception, le module DSS contient en moyenne 371,1 règles et le module CAP 1144,7 règles (soit un total de 1515,8 règles). Ce nombre est six fois plus élevé par rapport à l'étude de cas précédente. La complexité du processus d'analyse prédictive est donc approximativement six fois plus grande que la complexité évaluée qualitativement au chapitre 4.

Le processus d'expérimentation généralisé est plus complexe que le processus développé pour un espace de conception totalement ordonnée. Cette complexité additionnelle est liée à la correction des erreurs de classement le long de l'échelle de performance et à la sélection d'une configuration maximisant les chances de se rapprocher du comportement optimal. Ces opérations sont réalisées par des manipulations vectorielles (comparaison, tri, récupération et mise à jour d'informations). Les vecteurs sont courts (27 éléments) donc le processus d'expérimentation reste léger en ressources de calcul.

La complexité du processus de décision augmente avec la taille de la matrice de décision ($6 \times 6 \rightarrow 27 \times 27$). Le moteur cognitif supervise un équipement plus flexible donc il est légitime que les mécanismes cognitifs consomment plus de ressources de calcul. La méthode de prise de décision multi-critères utilisée permet néanmoins de maîtriser l'augmentation de la complexité puisqu'elle n'implique que des calculs matriciels optimisables à l'aide d'architectures matérielles dédiées.

Le processus d'apprentissage actualise la base de connaissance de deux systèmes évolutifs donc sa complexité augmente également. L'augmentation dépend des heuristiques appliquées. Elle reste très légère tant que le processus crée un nœud ou qu'il réduit le champ d'influence d'une règle erronée. Elle est beaucoup plus grande pour l'heuristique de généralisation des règles (dans l'état actuel des mécanismes). Cette procédure peut cependant être optimisée par des mécanismes avancés de gestion de la complexité (cf. chapitre 6).

En observant les courbes de la figure 5.22 à une échelle plus grande, nous remarquons que le nombre de règles peut baisser très légèrement de temps en temps suite aux mécanismes de

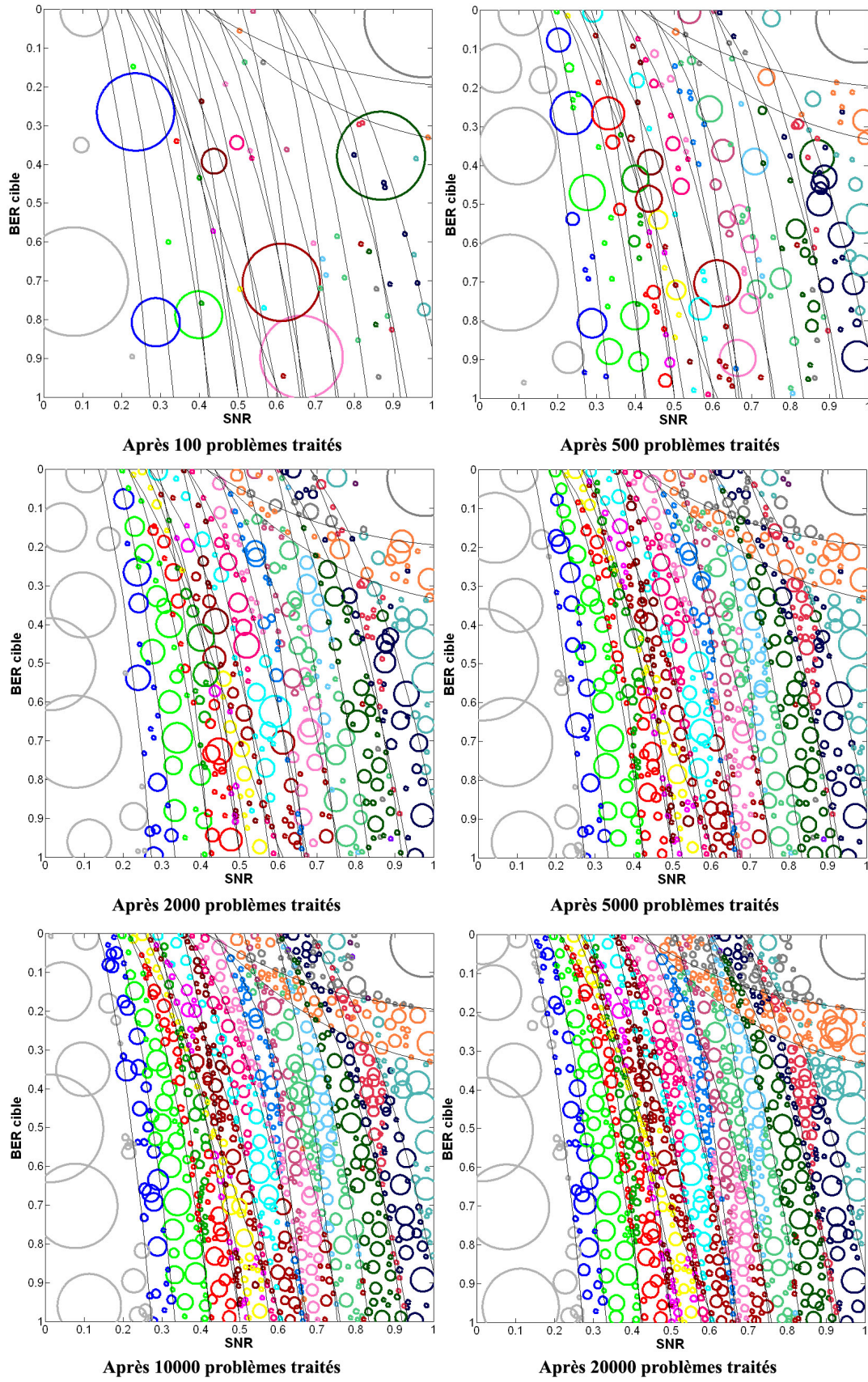


FIGURE 5.21 – Evolution de la carte d'adéquation avec l'expérience du moteur cognitif (FMP_3)

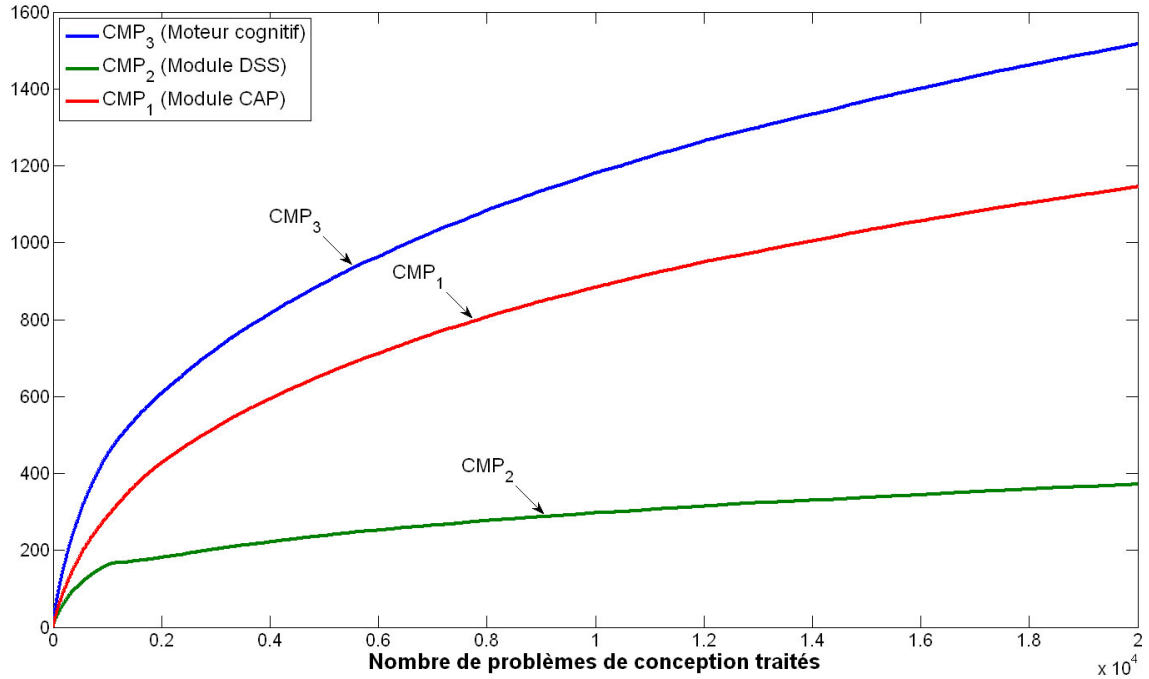


FIGURE 5.22 – Evolution du nombre de règles à l'intérieur des bases de connaissances (CMP_{1-3})
 $CMP_3 = CMP_2 + CMP_1$

désapprentissage. Ce phénomène s'observe surtout pour le module DSS oubliant les règles associées aux échelles de performance régulièrement mises en défaut par l'expérimentation. Il serait cependant nécessaire de développer des mécanismes plus efficaces afin de réduire encore plus la complexité des systèmes évolutifs. Le nombre de règles du module DSS semble en effet exagéré au regard de son espace de classification. Les courbes de performance se croisent un peu plus d'une quarantaine de fois pour une valeur du SNR comprise entre 0 et 20 dB et un BER cible inférieur à 10^{-2} . Ce chiffre est à comparer aux 371,1 règles du module DSS en fin de simulation. Nous pensons aussi que le nombre de règles aurait pu être réduit en choisissant une valeur plus petite pour R_{\min} . Nous pouvons observer en effet à la figure 5.10 que certaines courbes se croisent à des intervalles très proches (inférieur à la valeur de R_{\min} choisie) si l'on ne considère que la valeur du SNR. Le système connexionniste est alors obligé de compenser cette trop large granularité par un nombre plus grand de clusters pour maintenir des prédictions de bonne qualité.

Impact du moteur cognitif sur les indicateurs de performance de la radio

Le tableau 5.4 dresse un bilan comparatif des performances associées aux différentes radios considérées par notre étude de cas. Les conclusions sont sensiblement les mêmes que pour la première étude de cas donc nous renvoyons le lecteur aux commentaires du chapitre précédent.

Nous remarquons cependant que notre moteur cognitif arrive à maintenir globalement les mêmes chiffres qu'à l'étude de cas précédente en dépit des conditions de propagation plus difficiles. Ce résultat positif est lié à :

- l'introduction de l'égaliseur capable de compenser (au moins partiellement) les interférences inter-symboles pour les problèmes difficiles,
- l'ajout d'une nouvelle modulation permettant au moteur cognitif d'ajuster les compromis entre les objectifs secondaires de manière plus fine.

Radios non flexibles				BRD_1	BRD_2	BRD_3
<i>Label</i>	<i>Code correcteur</i>	<i>Modulation</i>	<i>Égaliseur</i>			
C_1	BCH(127, 92, 5)	BPSK	MLSE	78.3	0.57	477
C_2	BCH(127, 120, 1)	BPSK	MLSE	65.8	0.62	9
C_3	Aucun	BPSK	MLSE	59.6	0.60	2
C_4	BCH(127, 92, 5)	BPSK	Linéaire	72.9	0.53	487.1
C_5	BCH(127, 120, 1)	BPSK	Linéaire	59.2	0.56	19.1
C_6	Aucun	BPSK	Linéaire	52.6	0.53	12.1
C_7	BCH(127, 92, 5)	BPSK	Aucun	56.6	0.41	476
C_8	BCH(127, 120, 1)	BPSK	Aucun	12.1	0.11	8
C_9	Aucun	BPSK	Aucun	7.3	0.07	1
C_{10}	BCH(127, 92, 5)	QPSK	MLSE	63.2	0.92	479
C_{11}	BCH(127, 120, 1)	QPSK	MLSE	50.7	0.96	11
C_{12}	Aucun	QPSK	MLSE	44.5	0.89	4
C_{13}	BCH(127, 92, 5)	QPSK	Linéaire	56.1	0.81	487.1
C_{14}	BCH(127, 120, 1)	QPSK	Linéaire	43.0	0.81	19.1
C_{15}	Aucun	QPSK	Linéaire	36.5	0.73	12.1
C_{16}	BCH(127, 92, 5)	QPSK	Aucun	0	0	476
C_{17}	BCH(127, 120, 1)	QPSK	Aucun	0	0	8
C_{18}	Aucun	QPSK	Aucun	0	0	1
C_{19}	BCH(127, 92, 5)	8-PSK	MLSE	39.6	0.86	483
C_{20}	BCH(127, 120, 1)	8-PSK	MLSE	25.9	0.73	15
C_{21}	Aucun	8-PSK	MLSE	19.3	0.58	8
C_{22}	BCH(127, 92, 5)	8-PSK	Linéaire	31.4	0.68	487.1
C_{23}	BCH(127, 120, 1)	8-PSK	Linéaire	17.4	0.49	19.1
C_{24}	Aucun	8-PSK	Linéaire	10.8	0.32	12.1
C_{25}	BCH(127, 92, 5)	8-PSK	Aucun	0	0	476
C_{26}	BCH(127, 120, 1)	8-PSK	Aucun	0	0	8
C_{27}	BCAucun	8-PSK	Aucun	0	0	1
Radio cognitive idéale				78.3	1.27	104.7
Radio cognitive proposée				78.1	1.26	105.6

TABLE 5.4 – Comparaison des indicateurs de performance BRD_{1-3} pour des radios non flexibles et flexibles

Influence des connaissances expertes

Le graphe de robustesse de la figure 3.10 résulte d'une analyse par bloc sur les performances relatives des traitements disponibles. Il est possible cependant de disposer d'informations plus complètes sur la structure de l'espace de conception. L'observation des courbes de performance montre par exemple que les configurations sans égaliseur n'ont aucune chance de vérifier les contraintes si elles ne sont pas associées à la modulation la plus robuste. Plus concrètement, nous remarquons que les courbes associées à C_{16} , C_{17} , C_{18} , C_{25} , C_{26} et C_{27} sont toujours au-dessus des courbes associées à C_9 et C_{23} pour des valeurs de BER cible inférieures à 10^{-2} . Ces connaissances peuvent être profitables au moteur cognitif. Elles sont facilement encodables dans le graphe de robustesse. Il suffit pour cela d'ajouter un lien orienté de C_9 vers C_{15} et de C_{23} vers C_{15} . Ce graphe de robustesse plus instructif est illustré à la figure 5.23 en compagnie du graphe de robustesse initial (premier niveau de connaissance). L'analyse des courbes a été poussée encore plus avec le troisième niveau de connaissance, le graphe de robustesse se montrant beaucoup plus précis dans sa structuration de l'espace de conception.

Nous avons testé notre moteur cognitif dans les mêmes conditions en modifiant le niveau des connaissances embarquées. Les résultats marquants sont résumés dans le tableau 5.5. Nous donnons aussi l'évolution de quelques indicateurs de performance particulièrement affectées par la mise à disposition de connaissances plus instructives (figures 5.24 à 5.27).

Indicateur	Moyennées sur la simulation			En fin de simulation		
	<i>Niv. 1</i>	<i>Niv. 2</i>	<i>Niv. 3</i>	<i>Niv. 1</i>	<i>Niv. 2</i>	<i>Niv. 3</i>
CG_2	73,6	74,1	74,6	80,9	81,1	82,5
CG_3	97,0	97,3	97,5	—	—	—
CG_4	0,118	0,113	0,103	—	—	—
CG_5	1,80	1,76	1,72	1,51	1,50	1,46
CG_6	0,40	0,38	0,34	0,26	0,25	0,22
CG_{12}	0,091	0,066	0,051	0,044	0,030	0,020
CG_{13}	0,211	0,195	0,169	0,165	0,157	0,134
CG_{14}	0,188	0,146	0,142	0,138	0,108	0,103
CG_{15}	0,147	0,134	0,111	0,133	0,129	0,109
CG_{16}	0,132	0,093	0,093	0,111	0,079	0,080
FMP_1	85,4	85,8	86,1	87,6	87,9	87,9
FMP_2	78,0	78,3	79,3	82,7	82,7	83,9
$PSSE_1$	—	—	—	0,258	0,258	0,242
CMP_1	—	—	—	1521,8	1453,2	1379,0
CMP_2	—	—	—	373,1	313,9	262,1
CMP_3	—	—	—	1144,7	1139,3	1116,9

TABLE 5.5 – Influence des connaissances expertes (niveau 1, 2 ou 3) sur le comportement du moteur cognitif

Le moteur cognitif se comporte mieux s'il dispose d'une structuration plus forte de son espace de conception. Il fait confiance à ses connaissances plus rapidement pour limiter les risques d'expérimentation (*cf.* CG_2 et $PSSE_1$). Il approche le comportement optimal de plus près (*cf.* CG_3 et CG_4). Il écourte ses phases d'exploration plus rapidement (*cf.* CG_5) en prenant moins de mauvaises décisions (*cf.* CG_6 et fig. 5.24). Ses prédictions sont plus fiables (*cf.* FMP_1 , FMP_2

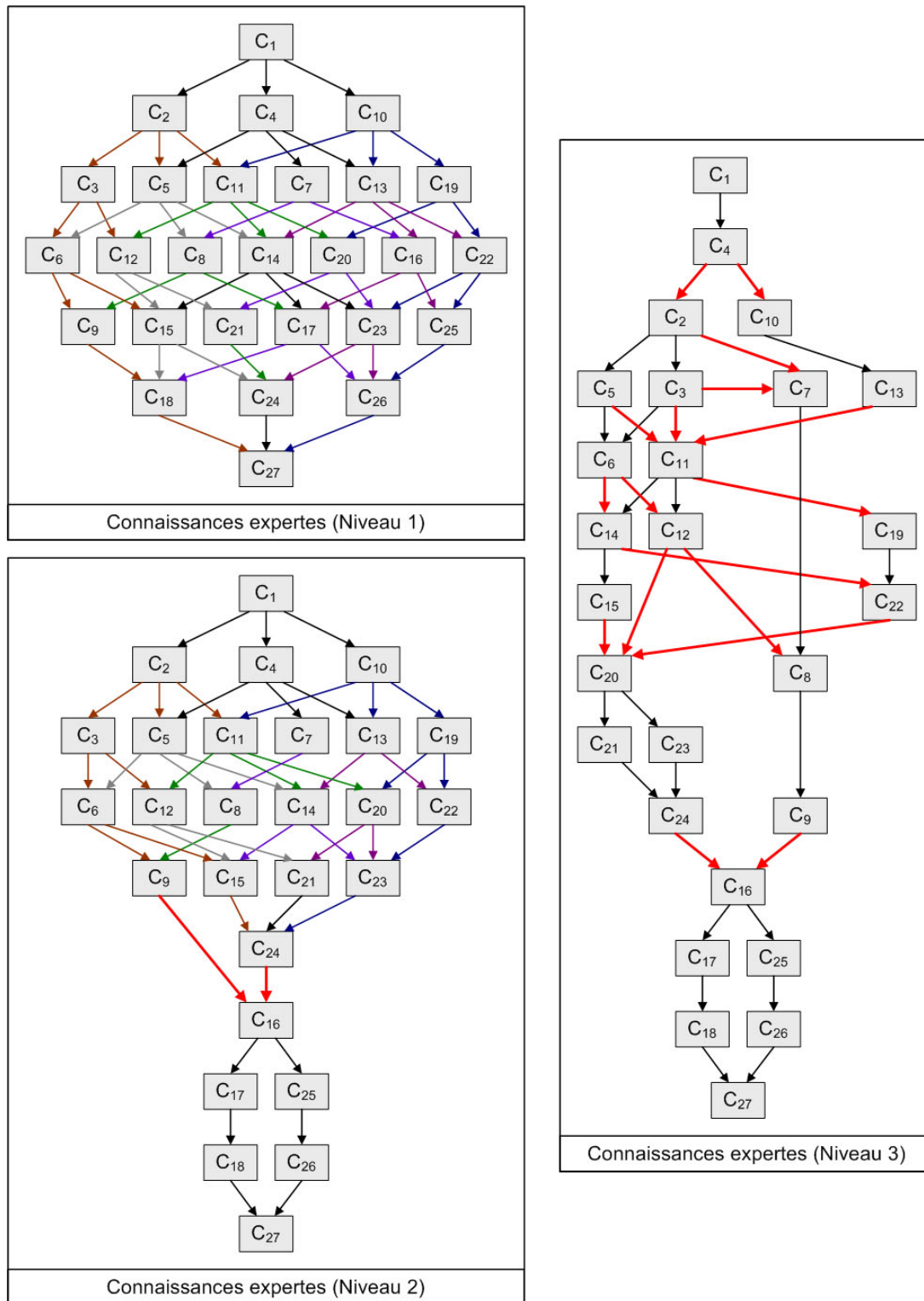


FIGURE 5.23 – Construction des graphes de robustesse à partir de connaissances expertes plus instructives sur l'espace de conception.

L'analyse préalable de l'espace de conception permet d'ajouter de nouveaux liens orientés mis en valeur en trait fort dans la figure.

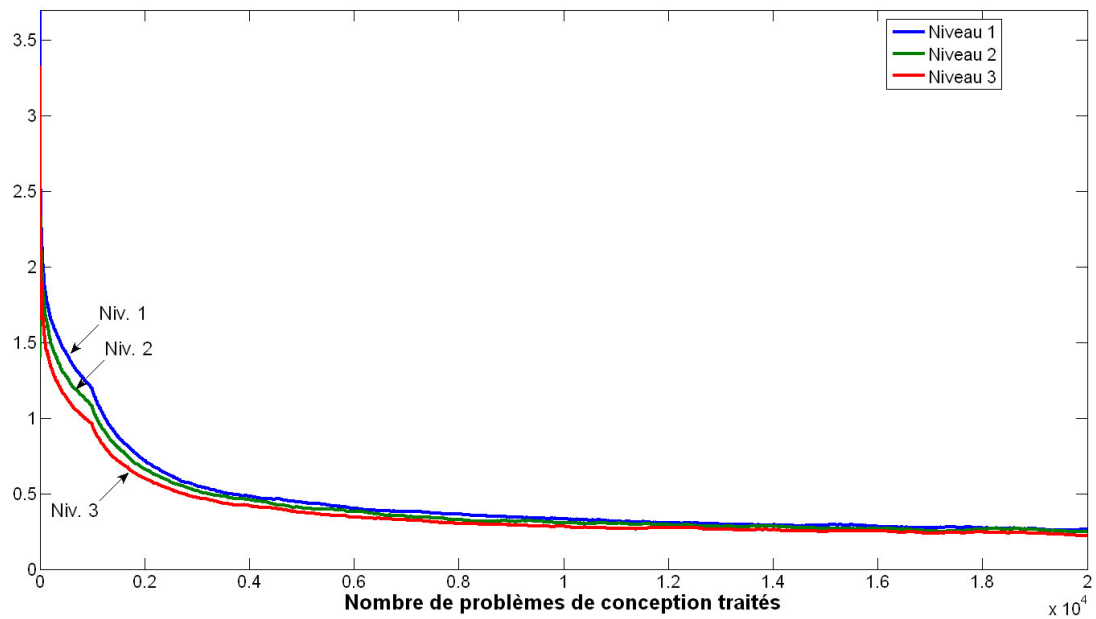


FIGURE 5.24 – Nombre moyenné de mauvaises décisions prises par expérience de conception en fonction de la qualité des connaissances expertes embarquées (CG_5)

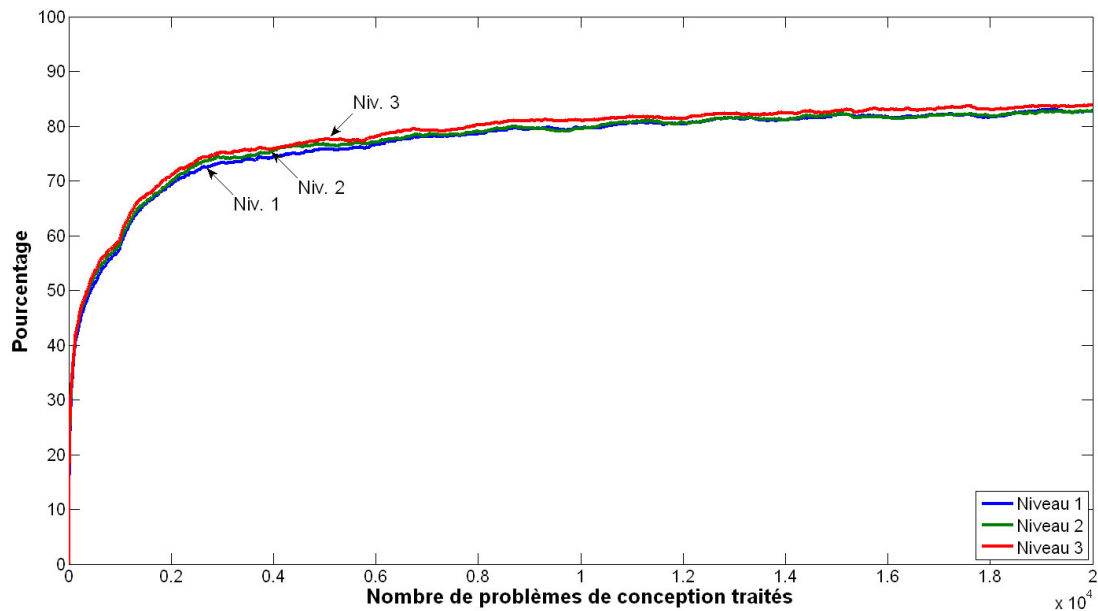


FIGURE 5.25 – Fiabilité des prédictions du module CAP en fonction de la qualité des connaissances expertes embarquées (FMP_2)

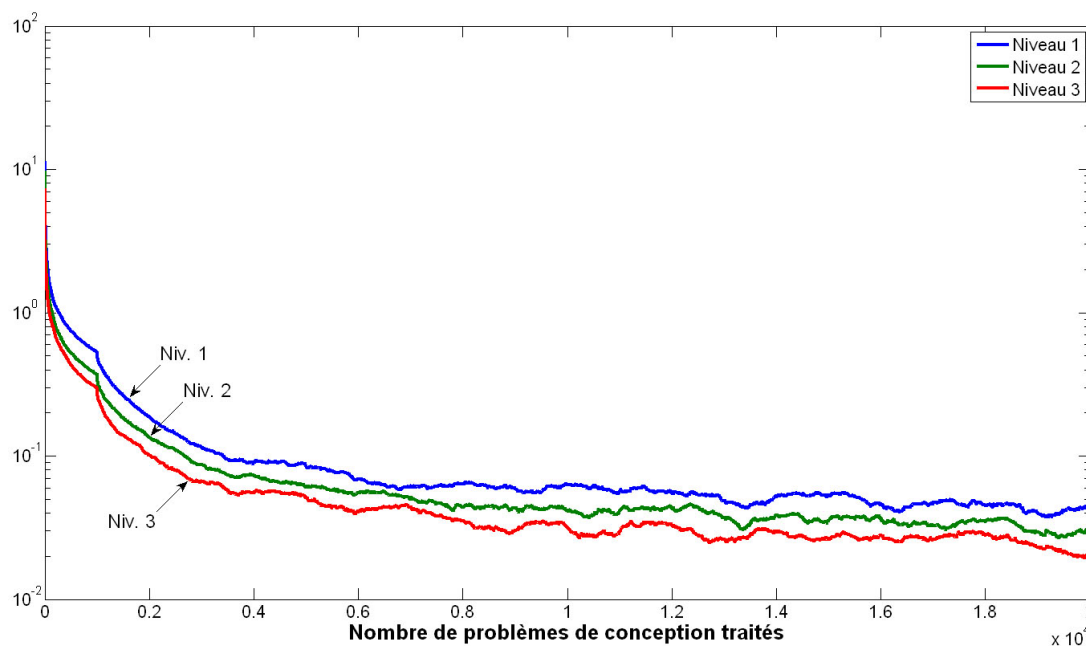


FIGURE 5.26 – Nombre moyenné d'incohérences identifiées par expérience de conception en fonction de la qualité des connaissances expertes embarquées (CG_{12})

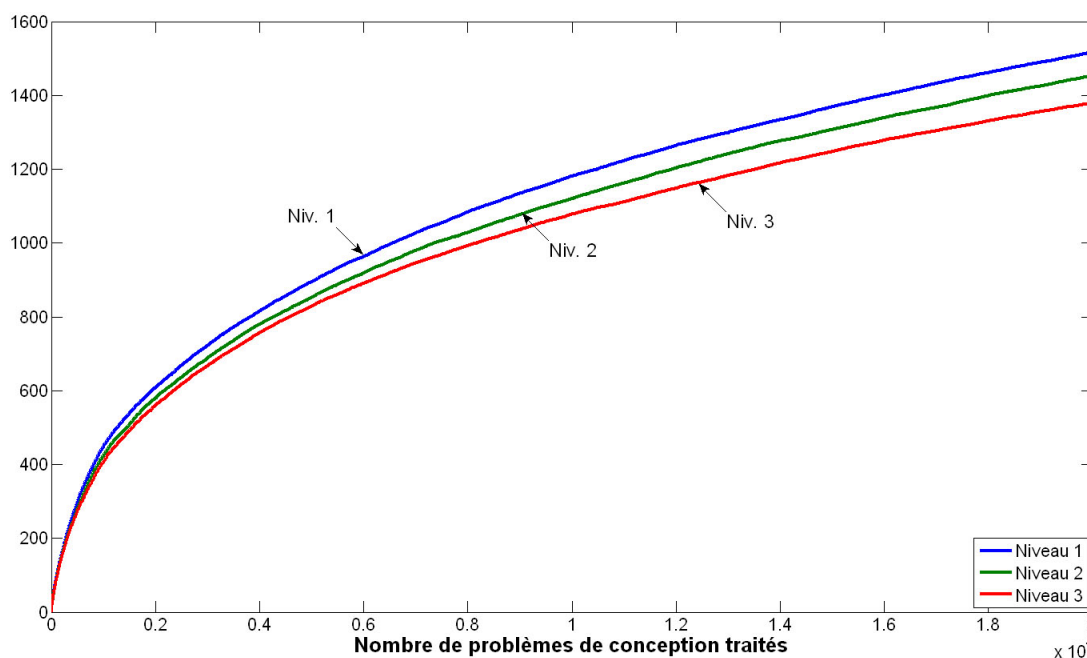


FIGURE 5.27 – Evolution de la taille des bases de connaissances du moteur cognitif en fonction de la qualité des connaissances expertes embarquées (CMP_3)

et fig. 5.25) car son espace est mieux caractérisé (cf. CG_{12} à CG_{16} et fig. 5.26). Enfin, les règles apprises sont plus pertinentes (notamment pour le module DSS) donc elles sont moins contredites par des expérimentations futures. Les bases de connaissances sont donc moins encombrées par des règles erronées ou sous-optimales (cf. CMP_1 à CMP_3 et fig. 5.27).

Notre moteur cognitif a donc l'avantage de pouvoir profiter de connaissances expertes pour améliorer son comportement tout en étant capable de s'en affranchir grâce à ses capacités d'apprentissage.

Généralisation à d'autres études de cas

Dans ce chapitre, nous avons testé notre moteur cognitif sur un espace de conception caractérisé par :

- 27 configurations disponibles,
- un modèle de canal particulier,
- une contrainte de performance donnée,
- 3 politiques d'optimisation.

Les mécanismes développés sont cependant génériques, le moteur cognitif étant capable de s'adapter à n'importe quelle carte de performance de par ses capacités d'apprentissage et d'exploration en ligne.

Le moteur cognitif apprend à identifier les fonctions discriminantes séparant les différentes régions de classification. Ces fonctions correspondent aux courbes de performance des configurations disponibles. Elles varient donc en fonction des opportunités de reconfiguration considérées, du canal modélisé et de l'indicateur de performance observé.

La figure 5.28 présente un exemple simplifié de l'évolution de la carte de performance lorsque le lien radio doit supporter un débit minimum plutôt qu'un taux d'erreurs maximum. Une fois la carte mémorisée, le moteur cognitif sera capable d'adapter l'efficacité spectrale de la radio en fonction du SNR estimé et du service sélectionné.

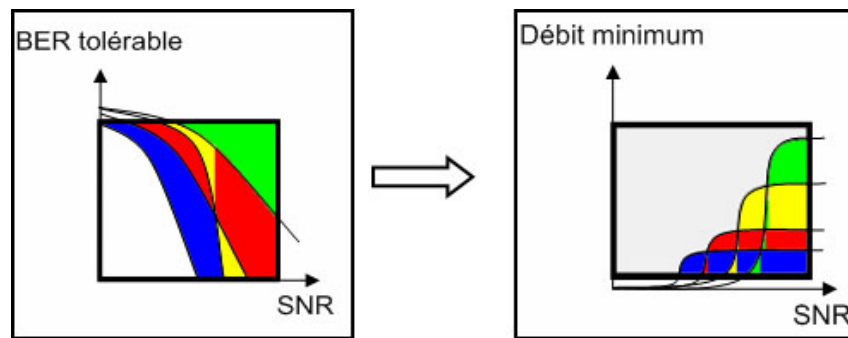


FIGURE 5.28 – Evolution de la carte de performance pour deux contraintes de performance différentes

Il est possible également d'augmenter le nombre de descripteurs de l'environnement radio afin de distinguer les modèles de canaux en fonction de leur sélectivité fréquentielle et temporelle. Le moteur cognitif aura alors à apprendre une carte à quatre dimensions au lieu de la carte à deux dimensions considérée dans cette thèse (cf. fig. 5.29).

Enfin, d'autres préférences d'optimisation sont envisageables à condition de développer les fonctions objectifs correspondantes.

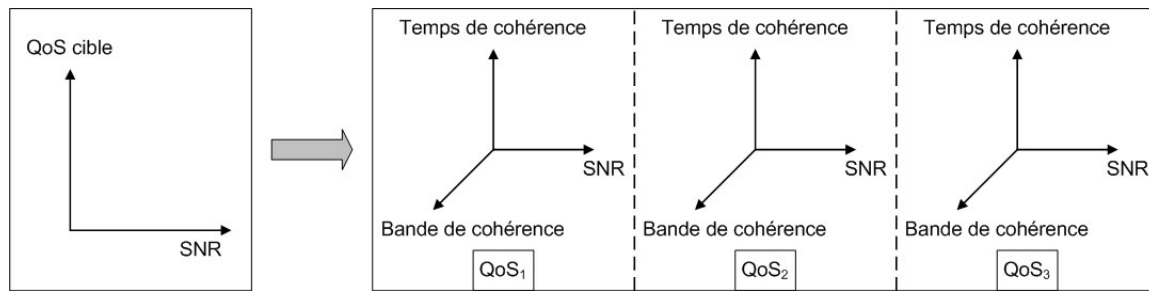


FIGURE 5.29 – Augmentation du nombre de descripteurs de l’environnement radio (apprentissage d’une carte à 4 dimensions)

Les systèmes connexionnistes sont des classifieurs universels ce qui leur permet de caractériser n’importe quel espace compact et borné du moment qu’ils disposent de suffisamment d’échantillons d’apprentissage [131]. La vitesse de progression du moteur cognitif dépend alors de la difficulté de l’espace de conception (nombre et superficie des régions de classification, nombre de points de croisement entre courbes de performance) et de la qualité des connaissances embarquées. Cette conclusion a été mise en évidence au travers des études de cas traitées dans le cadre de la thèse.

6

Moteur cognitif complet fondé sur les concepts proposés

Sommaire

6.1	Améliorations du moteur à l'aide de mécanismes cognitifs avancés .	138
6.1.1	Améliorations par des techniques existantes	138
6.1.2	Mécanismes proposés pour des améliorations supplémentaires	141
6.2	Gestion d'un environnement dynamique	142
6.2.1	Organisation structurelle du moteur cognitif	142
6.2.2	Organisation fonctionnelle du moteur cognitif	143
6.3	Gestion d'un espace de conception à complexité accrue	145
6.3.1	Prise en compte de plusieurs contraintes de performance simultanées . .	145
6.3.2	Augmentation de la flexibilité des équipements	146
6.4	Application des concepts à l'accès opportuniste au spectre	150

Les résultats présentés aux chapitres précédents témoignent de l'intérêt de l'approche proposée. Nous avons analysé notre solution en détail à l'aide d'un cadre évaluatif développé spécialement dans ce but. Nous avons montré que notre moteur cognitif apprenait à se rapprocher progressivement du comportement optimal en mettant ses connaissances à l'épreuve par une procédure d'essais/erreurs. Il caractérise son espace de conception en s'appuyant sur la mémoire de ses systèmes connexionnistes. Il limite les risques de ses expérimentations en contrôlant le dilemme exploitation/exploration à partir de la stratégie développée par son équipe d'automates d'apprentissage. Enfin, il intègre une intelligence haut-niveau pour guider sa recherche d'une configuration adaptée avec un souci d'efficacité et d'optimalité.

Des efforts de recherche/développement sont encore cependant nécessaires pour appliquer notre moteur cognitif aux problèmes réels de reconfiguration dynamique posés par les futures radios logicielles. Dans ce chapitre, nous proposons des mécanismes avancés pour améliorer les performances du moteur cognitif. Ces mécanismes restent encore à être validés expérimentalement mais ils offrent la perspective d'un moteur cognitif opérationnel, capable d'adapter le comportement de la radio aux variations de son contexte.

Dans un premier temps, nous présentons des méthodes pour optimiser le comportement du moteur cognitif et diminuer sa complexité. Nous indiquons ensuite les modifications à apporter pour lui permettre d'évoluer dans un environnement dynamique et ainsi suivre les variations de son contexte opérationnel. Nous précisons derrière les améliorations nécessaires pour traiter des

problèmes issus d'un espace de conception plus complexe (multiplication des objectifs, augmentation de la flexibilité des équipements). Nous concluons le chapitre en présentant les apports de nos travaux pour l'accès opportuniste au spectre.

6.1 Améliorations du moteur à l'aide de mécanismes cognitifs avancés

Notre moteur cognitif peut être enrichi sur de nombreux points afin d'améliorer la qualité de ses prises de décision, d'accélérer sa phase de recherche d'une configuration adaptée, de réduire ses besoins en mémoire et en ressources de calcul et d'augmenter sa vitesse d'adaptation.

Nous présentons dans un premier temps les solutions proposées par Kasabov *et al.* pour compenser les limitations liées aux systèmes connexionnistes évolutifs puis nous proposons nos propres suggestions d'optimisation.

6.1.1 Améliorations par des techniques existantes

Les systèmes connexionnistes évolutifs présentent des caractéristiques particulièrement intéressantes pour la prise de décision autonome dans un environnement variable. Le cadre d'application imaginé par Kasabov *et al.* parallélise d'ailleurs parfaitement le problème défini par une radio cognitive. Il est illustré à la figure 6.1. Les systèmes intelligents développés construisent leur base de connaissances en s'appuyant sur plusieurs modules connexionnistes (NNM : Neural Network Module) affectés chacun à une mission spécifique. Ils sont capables de prendre des décisions puis de les analyser à partir d'une simple évaluation critique de leur environnement. Ils savent aussi optimiser leur comportement afin d'améliorer la qualité et l'efficacité de leur réponse.

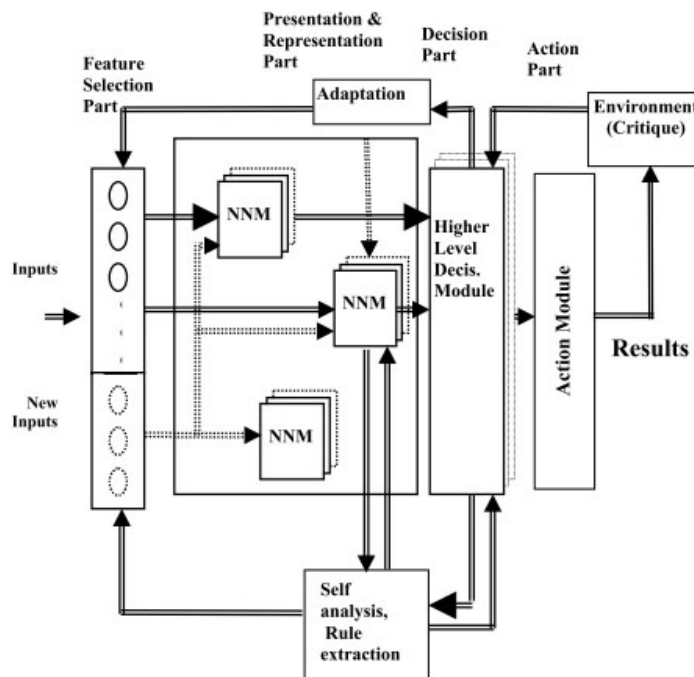


FIGURE 6.1 – Développement de systèmes intelligents basés sur des modules connexionnistes évolutifs (extrait de [146])

NNM : Neural Network Module

De nombreux mécanismes ont été proposés pour atteindre les objectifs d'adaptation, d'efficacité et d'autonomie de ces systèmes intelligents. Nous en recensons ici un certain nombre en renvoyant le lecteur intéressé aux références [131, 147, 145] pour plus de précisions.

Malgré leurs avantages, les systèmes connexionnistes évolutifs présentent certains défauts pouvant limiter leur intérêt pour le développement d'agents intelligents s'ils ne sont pas compensés suffisamment. Le premier défaut concerne la complexité du système lorsque la couche évolutive devient trop grande. Le second défaut est lié à la sensibilité des modules aux valeurs des paramètres spécifiées. Il existe des solutions pour réduire l'influence de ces deux limitations.

Gestion de la complexité

La couche évolutive doit être continuellement entretenue afin de préserver la qualité des décisions prises tout en minimisant la complexité des mécanismes d'inférence et d'apprentissage. Des règles peuvent être éliminées en fonction de leur âge si elles n'ont pas été activées depuis longtemps (*e.g.* règles de conception associées à un environnement radio caractéristique d'un lieu de vacances fréquenté). Il est possible également d'agréger les règles d'influence similaire afin de généraliser l'association mémorisée tout en diminuant le nombre de nœuds pour la représenter³⁷. La réduction du nombre de nœuds permet de diminuer le nombre de calculs nécessaires au parcours de la base de connaissance donc d'augmenter la vitesse des mécanismes appliqués. Les procédures d'élimination et d'agrégation sont donc essentielles pour la garantie d'une prise de décision temps réel sur le long terme.

La complexité du système peut être réduite encore plus en réalisant une allocation intelligente des nœuds à l'intérieur de la couche évolutive. La stratégie employée jusqu'à maintenant se contente d'insérer les nouveaux nœuds à la suite des nœuds déjà créés. Il serait plus intéressant de préserver la proximité spatiale des nœuds dans l'espace d'entrée. Les processus d'inférence et d'apprentissage pourraient alors parcourir la base de connaissances efficacement en orientant leurs calculs vers la région de l'espace de conception visitée.

Optimisation des paramètres

Le comportement des systèmes connexionnistes évolutif est dicté par plusieurs paramètres qu'il serait préférable d'adapter en ligne pour optimiser la qualité du modèle développé. Pour le modèle ECF, nous pouvons considérer le nombre de fonctions de membre associées à chaque variable d'entrée, leur position, leur forme, la valeur des rayons minimal et maximal ainsi que le nombre de règles utilisées pour l'inférence. Pour la mise à jour des automates d'apprentissage, il faudrait ajouter la valeur du rayon de voisinage et le paramètre de récompense/pénalité. Le comportement défensif peut aussi être ajusté pour rendre l'expérimentation plus efficace.

L'adaptation des paramètres peut se faire en calculant en ligne des statistiques sur les données d'entrée. La distribution des données traitées peut servir, par exemple, à augmenter la pertinence de la quantification floue réalisée par la seconde couche du système. Les fonctions de membre seraient alors affectées aux régions les plus denses de l'espace afin de pouvoir les caractériser plus finement. Le système peut aussi décider de créer une nouvelle fonction de membre si le problème n'est pas reconnu à un degré suffisamment fort par la couche de quantification (*e.g.* le degré d'activation maximum doit être au moins supérieur à 0.6). Il faut faire attention cependant à ne pas trop multiplier le nombre de fonctions de membre car cela augmente la dimensionnalité de

37. La procédure d'agrégation peut même appliquer un principe inspiré des machines à vecteur de support [148, 149] et laisser des nœuds de garde aux frontières de décision afin de maintenir les capacités discriminatives du système de classification.

l'espace d'entrée donc la complexité des opérations nécessaires au calcul de similarité entre les problèmes [147].

Le moteur cognitif peut aussi ajuster ses paramètres à travers une analyse introspective de son comportement. Lors de la présentation des résultats, nous avons notamment proposé des heuristiques pour optimiser la mise à jour de la stratégie stochastique d'exploration. Nous avons vu aussi qu'il serait préférable de contrôler le dilemme exploration/exploitation après chaque décision lorsque le moteur cognitif a acquis une certaine expérience sur son espace de conception.

Les paramètres peuvent enfin être optimisés par des méthodes évolutionnaires (*e.g.* algorithmes génétiques, stratégies d'évolution). Le moteur cognitif fait alors évoluer une population de systèmes connexionnistes sur plusieurs générations. Il laisse les différents systèmes apprendre à partir des interactions de la radio avec son environnement. Il les évalue après un certain temps afin de garder les meilleurs systèmes et leurs enfants pour la prochaine génération. Ce processus d'adaptation peut être appliqué en ligne [145] mais il reste exigeant en ressources de calcul. Il semble donc plus approprié pour les stations de base non contraintes en énergie. Les stations de base pourraient aussi décider d'intégrer les radios connectées dans un processus d'évolution distribué en leur affectant une petite population de systèmes connexionnistes à faire évoluer.

Diversité des modèles connexionnistes évolutifs

Il existe d'autres modèles connexionnistes pouvant s'avérer plus intéressants que le modèle ECF (*e.g.* EFuNN, ESOM, ECMC). Il serait donc utile de réaliser une étude comparative afin d'identifier le modèle assurant le meilleur compromis entre la qualité des prédictions, la complexité des mécanismes et leur vitesse de convergence.

Un modèle complexe peut s'avérer préférable à un modèle plus simple si sa couche évolutive continue à garder une taille raisonnable après un grand nombre de problèmes traités. Le modèle ECF a l'inconvénient, par exemple, de ne pas modifier les positions des associations mémorisées. Il se contente de modifier le rayon d'influence des règles, voire de les effacer. Il serait préférable de pouvoir déplacer le centre des règles vers les centres de gravité des données représentées. Le moteur cognitif pourrait ainsi résumer ses connaissances au travers d'un faible nombre de règles particulièrement pertinentes pour la résolution des problèmes traités.

De même, l'espace de classification sera plus efficacement représenté si le modèle utilisé mémorise des clusters prenant une forme plus en adéquation avec les différentes régions de décision. Les clusters ellipsoïdaux semblent par exemple plus adaptés à notre espace de conception que les clusters circulaires utilisés jusque là.

Mise en place d'une mémoire à court terme

Les systèmes connexionnistes peuvent intégrer une mémoire à court terme en ajoutant une deuxième couche évolutive mémorisant les caractéristiques temporelles des données. Les associations temporelles apprises permettent alors de renforcer l'activation des règles liées dans le temps. Le moteur cognitif peut donc optimiser ses réponses à une suite de problèmes fréquemment connectés (*e.g.* trajet de l'utilisateur de son domicile à son travail).

Coopération de plusieurs modules connexionnistes

Dans un contexte multi-modules, il est possible de créer des connexions entre plusieurs bases de connaissance si leurs règles sont régulièrement activées simultanément. Cette caractéristique

présente un intérêt pour l'optimisation inter-couches entre modules affectés à des tâches cognitives différentes (*e.g.* optimisation du récepteur, accès opportuniste au spectre).

6.1.2 Mécanismes proposés pour des améliorations supplémentaires

Le comportement du moteur cognitif peut être amélioré par une phase de consolidation des connaissances activée lorsque le terminal se raccorde à une source d'énergie (similaire à la phase de sommeil pour l'homme). Le moteur cognitif profite de cette période pour lancer des algorithmes d'optimisation gourmands en ressources de calcul. Les connaissances peuvent alors être consolidées de plusieurs manières.

Le moteur cognitif peut calculer les frontières de décision de ses systèmes évolutifs puis réorganiser leur base de connaissances pour qu'elles assurent une couverture aussi parcimonieuse que possible de l'espace de classification.

Les connaissances peuvent être structurées plus efficacement en appliquant un algorithme de clustering hiérarchique. L'algorithme analyse les bases de connaissances développées pour décomposer les plus grandes en plusieurs bases plus petites. Cette opération diminue la complexité des processus d'analyse prédictive et d'apprentissage en facilitant l'accès et l'exploitation des connaissances mémorisées. Par exemple, un espace représenté par 400 règles peut être décomposé en 4 espaces de 100 règles afin de réduire le nombre de calculs de distance. Dans le premier cas, le moteur cognitif doit calculer la distance du problème aux 400 règles (400 calculs). Dans le second cas, il sélectionne une base de connaissances parmi les quatre développées puis il calcule la distance du problème aux 100 règles de la base sélectionnée (104 calculs).

Les mécanismes d'inférence seront plus efficaces si les systèmes évolutifs établissent des liens entre les neurones voisins pour indiquer leur proximité. Le moteur cognitif peut alors profiter de ces liens orientés pour se déplacer plus rapidement dans les cartes prédictives et ainsi limiter les calculs tout en augmentant la réactivité du moteur cognitif.

La consolidation des connaissances peut être guidée par des heuristiques expertes déduites de notre compréhension de l'espace de conception et du comportement des mécanismes embarqués. L'espace de conception a l'avantage d'être facilement interprétable dans notre modèle de prise de décision. Les frontières de décision correspondent aux courbes de performance des configurations disponibles et il existe une grande cohérence entre les problèmes de conception similaires. Les échelles de performance ne peuvent être complètement bouleversées en passant d'une règle à sa règle voisine. Les rangs de plus haute compatibilité ne peuvent pas prendre de valeurs trop éloignées non plus. Le processus de consolidation peut donc repérer des incohérences dans les bases de connaissances en comparant les informations stockées par un neurone avec les informations d'un neurone proche. Il augmentera légèrement la probabilité d'exploration locale s'il découvre des transitions trop brutales entre règles similaires. Il indiquera aussi au processus d'expérimentation quelles sont les meilleures opportunités d'exploration. Ces recommandations sont obtenues par analyse du voisinage de chaque règle. Le processus de consolidation cherche alors à estimer la probabilité de succès des configurations supposées incompatibles (*i.e.* de rang supérieur au HCR prédit). Il observe leur adéquation pour des problèmes similaires. Il étudie aussi les mémoires d'exploration locales afin de prioriser plus finement les configurations à expérimenter.

Le moteur cognitif peut enfin augmenter sa vitesse d'adaptation (éventuellement jusqu'au rythme paquet) en construisant une table de correspondance pour chaque situation typique (*e.g.* service voix à domicile) afin de stocker les seuils de décision locaux. Ces seuils sont alors simplement déduits des règles régulièrement activées par les situations considérées.

6.2 Gestion d'un environnement dynamique

Dans son état actuel, notre moteur cognitif n'est pas encore opérationnel pour évoluer dans un environnement dynamique. Des mécanismes supplémentaires sont en effet nécessaires pour tenir compte de la variabilité du contexte opérationnel de la radio. Nous revisitons donc une nouvelle fois les organisations structurelle et fonctionnelle de notre moteur cognitif pour indiquer les modifications à apporter par rapport aux mécanismes décrits au chapitre 5.

6.2.1 Organisation structurelle du moteur cognitif

La figure 6.2 présente la nouvelle organisation structurelle du moteur cognitif. Nous avons ajouté un code supplémentaire en complément des cadres en trait plein et en pointillé³⁸. Dans le même esprit, l'alternance de traits courts/trait longs représente les informations soumises aux incertitudes de mesure (*e.g.* modèle du canal, indicateurs de performance). Les incertitudes viennent ici de la variabilité stochastique du canal et de la précision limitée des capteurs utilisés.

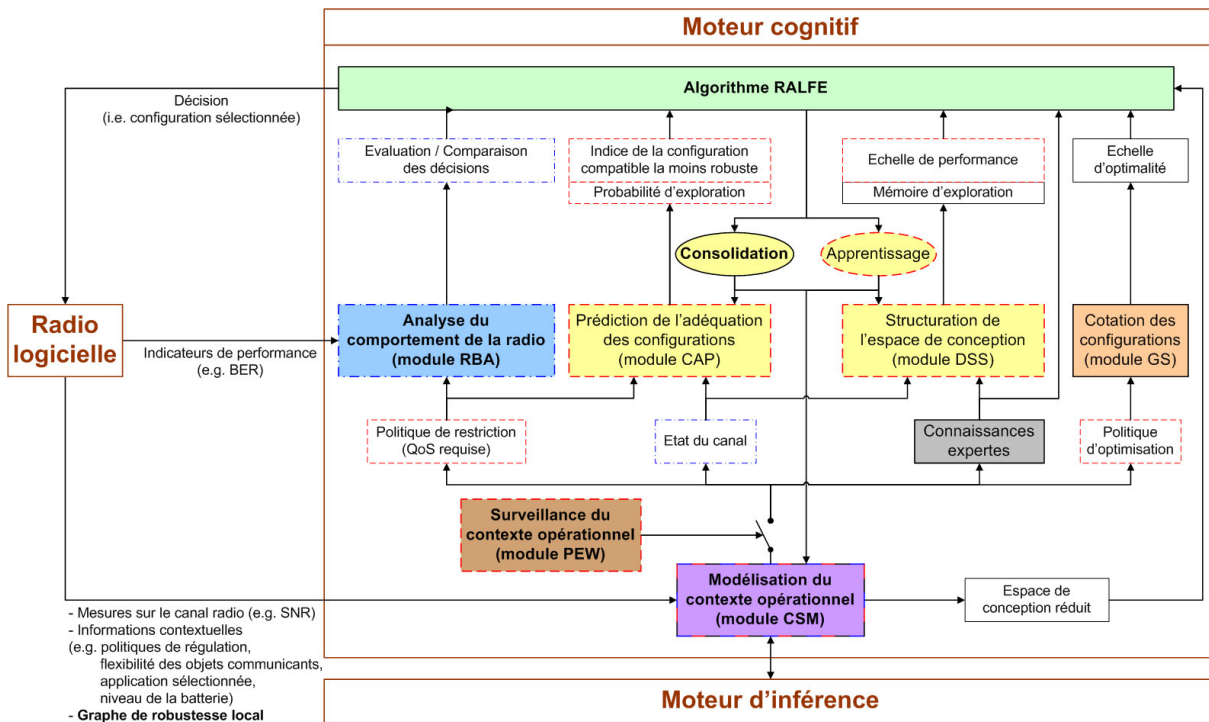


FIGURE 6.2 – Organisation structurelle du moteur cognitif avancé

Le module CSM est associé aux trois codes car il réalise une modélisation experte et prédictive du contexte en présence d'incertitudes. L'analyse experte du contexte opérationnel permet d'identifier les politiques de conception (optimisation et restriction) définissant le problème à solutionner. Elle est complétée par une analyse prédictive permettant :

- d'estimer les performances maximales atteignables pour l'environnement radio considéré,
- de modéliser plus finement les préférences de l'utilisateur.

Le module CSM doit enfin tenir compte des incertitudes de mesures lors de sa modélisation de l'environnement radio.

38. Pour rappel, le trait plein fait référence aux informations et aux connaissances fiables alors que les pointillés sont liés aux incertitudes de prise de décision causées par les erreurs de prédiction.

Le moteur cognitif compte ensuite sur le module PEW (Problem Evolution Watcher) pour suivre les variations du contexte opérationnel. Le module observe l'évolution du problème de conception pour déterminer quand une reconfiguration s'avère nécessaire ou quand il est préférable de maintenir la configuration actuelle pour ne pas déclencher les mécanismes de prise de décision (prédiction des performances, recherche de la configuration optimale et expérimentation) inutilement. Son comportement sera rapidement présenté à la section 6.2.2.

Les deux dernières modifications structurelles concernent des améliorations déjà introduites. Les connaissances mémorisées par les modules prédictifs (CAP, DSS et CSM) sont optimisées par un apprentissage en ligne renforcé par une phase hors-ligne de consolidation des connaissances. Les connaissances expertes peuvent être mises à jour grâce au support de l'infrastructure réseau collectant les résultats d'expérimentation des radios connectées pour construire un graphe de robustesse local qui soit fiable et informatif. Ce graphe peut alors être diffusé en début de communication pour diminuer les risques de décision et d'expérimentation.

6.2.2 Organisation fonctionnelle du moteur cognitif

Le moteur cognitif intègre deux nouveaux processus pour l'analyse des informations contextuelles et l'identification des problèmes de conception à résoudre. L'algorithme de supervision modifié est illustré à la figure 6.3.

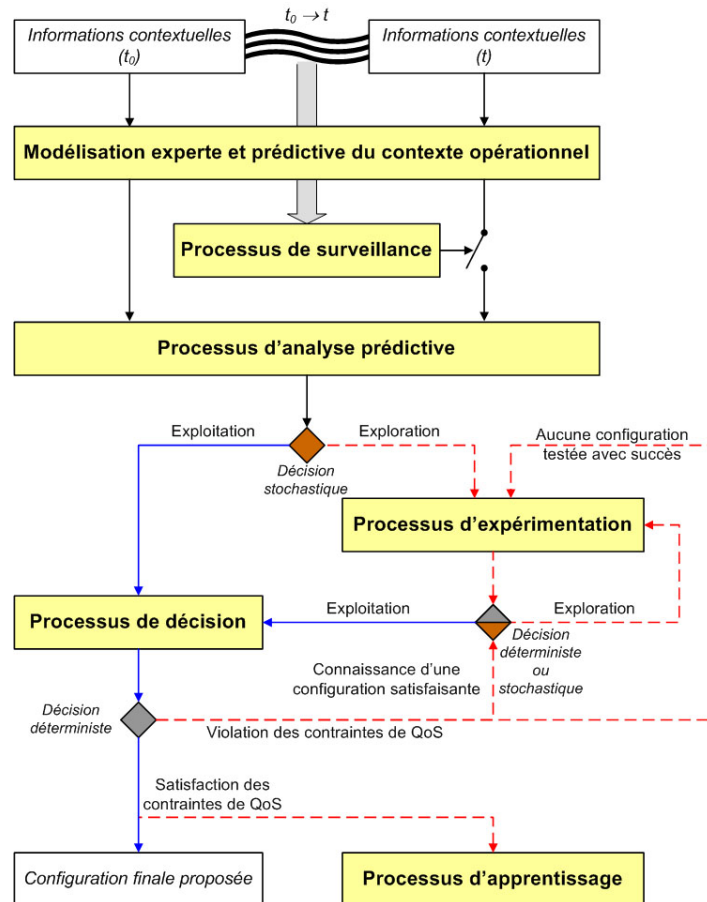


FIGURE 6.3 – Organisation fonctionnelle du moteur cognitif

Le processus de modélisation du contexte opérationnel applique les principes énoncés à la section 3.1. Les objectifs de conception sont déduits par une approche experte exploitant la logique événementielle du moteur d'inférence. Le processus identifie les contraintes de conception imposées en analysant les politiques de régulation, le profil de l'application sélectionnée et les profils des équipements engagés dans la communication. Il tient compte des préférences de l'utilisateur, des caractéristiques de l'application (*e.g.* temps réel ou non) et des contraintes énergétiques de l'équipement pour dériver la politique d'optimisation guidant la recherche d'une configuration particulièrement adaptée.

Le moteur cognitif peut utiliser un système connexionniste supplémentaire pour mémoriser la QoS atteignable en fonction du canal radio utilisé. La conception du lien radio est guidée au départ par des contraintes de performance représentatives d'une QoS moyenne. Le moteur cognitif diminue ensuite les contraintes s'il n'arrive pas à identifier de configurations compatibles. Il peut aussi chercher à appliquer un code source de meilleure qualité (*e.g.* vidéo haute définition) s'il évolue dans la portion haute de l'échelle de performance. Il sait dans ce cas qu'il évolue dans un environnement radio favorable (le nombre de configurations compatibles est élevé) donc il peut augmenter les exigences de QoS. Le nouveau système connexionniste doit contenir autant de classes que de niveaux de QoS disponibles. Le processus d'expérimentation n'a alors besoin que d'une légère modification pour mettre en place ces nouvelles procédures de test et d'analyse. Le processus d'apprentissage pourra ensuite exploiter les informations déduites de l'exploration pour entraîner le système connexionniste à l'aide d'un algorithme similaire aux algorithmes présentés dans cette thèse.

La nature stochastique de l'environnement radio requiert des outils d'analyse statistique pour moyenniser les mesures réalisés par les capteurs. Le moteur cognitif suit les variations temporelles du canal liées à l'effet de masque mais il doit aussi savoir réagir rapidement aux événements imprévus (*e.g.* apparition d'un interférent). Il est donc nécessaire de réaliser une moyenne glissante des mesures sur des fenêtres de longueur différente. La première fenêtre évolue au rythme des variations grande échelle. Sa longueur dépend du temps de cohérence du canal estimé à partir des échantillons reçus. La fenêtre sera donc étroite pour des canaux à évanouissements rapides puisque les différents états du canal sont visités sur une période de temps plus courte. La fenêtre temporelle peut devenir beaucoup plus large pour des canaux à évanouissements lents. Sa longueur sera déterminée à partir d'heuristiques qu'il reste à développer. La seconde fenêtre sera toujours plus courte que la première afin de détecter des variations rapides du canal (*e.g.* passage derrière un immeuble).

L'analyse du comportement de la radio doit elle aussi tenir compte de la variabilité des mesures réalisées. Le renforcement ne peut plus être calculé à partir d'une simple comparaison de la valeur mesurée avec la contrainte de performance imposée. Le module RBA doit plutôt utiliser des tests d'hypothèse (*e.g.* tests de Student) pour estimer les performances des configurations choisies et comparer leurs performances relatives.

Le processus de surveillance observe l'évolution du contexte opérationnel pour identifier quand une reconfiguration s'avère nécessaire. Il s'appuie sur la logique événementielle du moteur d'inférence pour :

- prendre en charge de nouvelles politiques de régulation,
- réagir à un changement d'application,
- déterminer si la configuration actuelle peut être implémentée sur un terminal moins flexible se connectant au réseau supervisé.

Il suit les variations du canal radio en naviguant à l'intérieur des cartes de performances développées par les systèmes connexionnistes (modules CSM, DSS et CAP). Il se déplace efficacement

dans ces cartes en profitant des connexions latérales introduites entre les neurones voisins de la couche évolutive (*cf.* sec. 6.1.2). Cette technique réduit le nombre de calculs de distance nécessaires à l'identification du voisinage du problème. Le moteur cognitif peut donc surveiller son environnement sans impacter l'autonomie énergétique du terminal. Il peut même avoir une attitude proactive s'il intègre un algorithme pour calculer la trajectoire de l'utilisateur dans ces cartes de performance.

Le moteur cognitif peut aussi profiter de la nature dynamique de l'environnement pour étaler ses prises de risque dans le temps. Le processus d'expérimentation peut décider d'arrêter ses tentatives d'exploration pour ne pas dégrader la QoS trop longtemps. Le processus de décision est ensuite activé pour revenir à des performances appréciables. Plus tard, l'algorithme de supervision peut décider de revenir à l'expérimentation afin de mieux caractériser la région visitée. Les erreurs de décision ne sont plus aussi rapprochées dans le temps donc elles affectent moins l'expérience de l'utilisateur.

6.3 Gestion d'un espace de conception à complexité accrue

Dans leur état, nos mécanismes sont transposables à de nombreux problèmes de reconfiguration dynamique tant qu'ils restent définis sur des espaces de conception relativement simples. Il serait intéressant, par exemple, de tester notre moteur cognitif sur le scénario de communication introduit dans [93]. Les auteurs considèrent un terminal flexible exploitant différentes techniques de diversité spatiale ainsi que plusieurs schémas de modulation et de codage. Le moteur cognitif doit alors choisir la combinaison la plus en adéquation avec le canal radio observé afin de maximiser l'efficacité spectrale du lien radio tout en maintenant des performances minimales. Son espace de conception a une complexité du même ordre de grandeur que l'espace défini par notre scénario. Les auteurs ne considèrent en effet que deux objectifs en compétition et le nombre d'opportunités de reconfiguration est limité à 45 configurations disponibles. Nos mécanismes cognitifs sont donc susceptibles de se comporter efficacement face aux problèmes définis sur cet espace. Les modifications à apporter sont mineures puisqu'elles concernent uniquement la définition des espaces d'entrée/sortie du moteur cognitif et de ses modules internes.

Pour des espaces plus larges et plus complexes, il est nécessaire de généraliser nos mécanismes et de proposer de nouvelles solutions techniques. Nous proposons dans cette section quelques réflexions allant dans ce sens.

6.3.1 Prise en compte de plusieurs contraintes de performance simultanées

Lors de la validation de nos mécanismes cognitifs, nous avons imposé une contrainte de performance uniquement sur le nombre d'erreurs de transmission tolérables. Le taux d'erreurs binaires est un indicateur de performance couramment utilisé pour définir les exigences de QoS. Il peut cependant être complété par d'autres contraintes accompagnant le débit et/ou la latence de la communication. Le moteur cognitif doit alors vérifier des contraintes sur plusieurs dimensions. Il lui suffit dans ce cas de développer un nouveau modèle prédictif pour toute contrainte supplémentaire. Chaque contrainte se voit alors associer :

- un module DSS pour l'identification de l'échelle de performance,
- un module CAP pour la prédiction de la configuration compatible la moins robuste,
- un graphe de robustesse mémorisant les relations d'ordre entre les alternatives.

Le processus d'analyse prédictive active les modules concernés par l'application considérée (une application non temps réel impose rarement une contrainte sur la latence de la communi-

cation). Chaque modèle prédictif produit alors son ensemble de configurations compatibles. Le processus de décision n'a plus qu'à rechercher la configuration la plus adaptée dans l'ensemble fusionné des configurations compatibles avec toutes les contraintes. Le processus d'expérimentation doit par contre être généralisé car il doit naviguer sur plusieurs échelles de performance pour identifier les configurations les plus attrayantes à explorer.

6.3.2 Augmentation de la flexibilité des équipements

Le nombre de configurations disponibles augmente rapidement avec la flexibilité des équipements. Il suffit d'ajouter une option de reconfiguration supplémentaire aux trois blocs supervisés (modulation, codage canal, égalisation) pour passer de $3 \times 3 \times 3 = 27$ configurations à $4 \times 4 \times 4 = 64$ configurations disponibles. Si le moteur cognitif contrôle la reconfiguration d'un bloc supplémentaire (*e.g.* diversité spatiale) caractérisé par trois techniques disponibles, ce nombre passe à $3 \times 64 = 192$ configurations.

Le nombre de règles de conception augmente avec le nombre de configurations donc le moteur cognitif aura besoin de plus de mémoire et de ressources de calcul. Les échelles de performance mémorisées par le module DSS seront représentées par des vecteurs plus longs et le module CAP intégrera autant de classes que de configurations disponibles. Ce phénomène est néanmoins commun à toutes les solutions. Pour l'approche experte, le moteur d'inférence doit parcourir une base de connaissance plus grande avant d'identifier les règles à appliquer. Les chromosomes développés pour l'approche exploratoire seront plus longs donc le processus d'évaluation sur plusieurs générations sera plus coûteux en ressources de calcul. Le modèle développé par l'approche prédictive sera de la même manière plus complexe.

Les exigences croissantes du moteur cognitif peuvent être contrôlées par les procédures d'optimisation introduites à la section 6.1. Il sera nécessaire notamment de décomposer les bases de connaissances par clustering hiérarchique pour "zoomer" sur les régions complexes de l'espace de conception caractérisées par un grand nombre de surfaces de décision. D'autres solutions peuvent être considérées. Nous en proposons ici quelques unes.

Décomposition de l'espace de conception par fonctionnalité de traitement du signal

Il est possible d'affecter un système prédictif à chaque bloc reconfigurable afin de réduire la complexité de l'espace de classification à caractériser. Cette solution est illustrée à la figure 6.4. Nous y avons reproduit l'espace de classification du module CAP pour notre seconde étude de cas selon quatre points de vue. La figure (a) représente l'espace de classification du système prédictif chargé de prédire l'adéquation des 27 configurations aux problèmes posés. Les figures (b) à (d) représentent les espaces de classification pour les trois blocs reconfigurables traités séparément³⁹. Les régions de classification sont plus grandes une fois l'espace décomposé. Les systèmes évolutifs peuvent donc couvrir leur espace respectif avec un nombre de règles plus faible comparé à la solution monolithique utilisée jusque là.

Cette remarque doit cependant être pondérée car le module CAP a besoin de créer une représentation de l'espace de conception pour chaque bloc. Le nombre de règles sera donc plus élevé au total avec les trois bases de connaissances développées. Cette solution présente des avantages malgré tout :

39. Un système connexionniste supplémentaire est nécessaire pour distinguer les problèmes sans solution des problèmes résolus par la configuration la plus robuste. Ce système est appelé uniquement lorsque les systèmes prédictifs recommandent l'option la plus robuste pour chacune des fonctionnalités du signal.

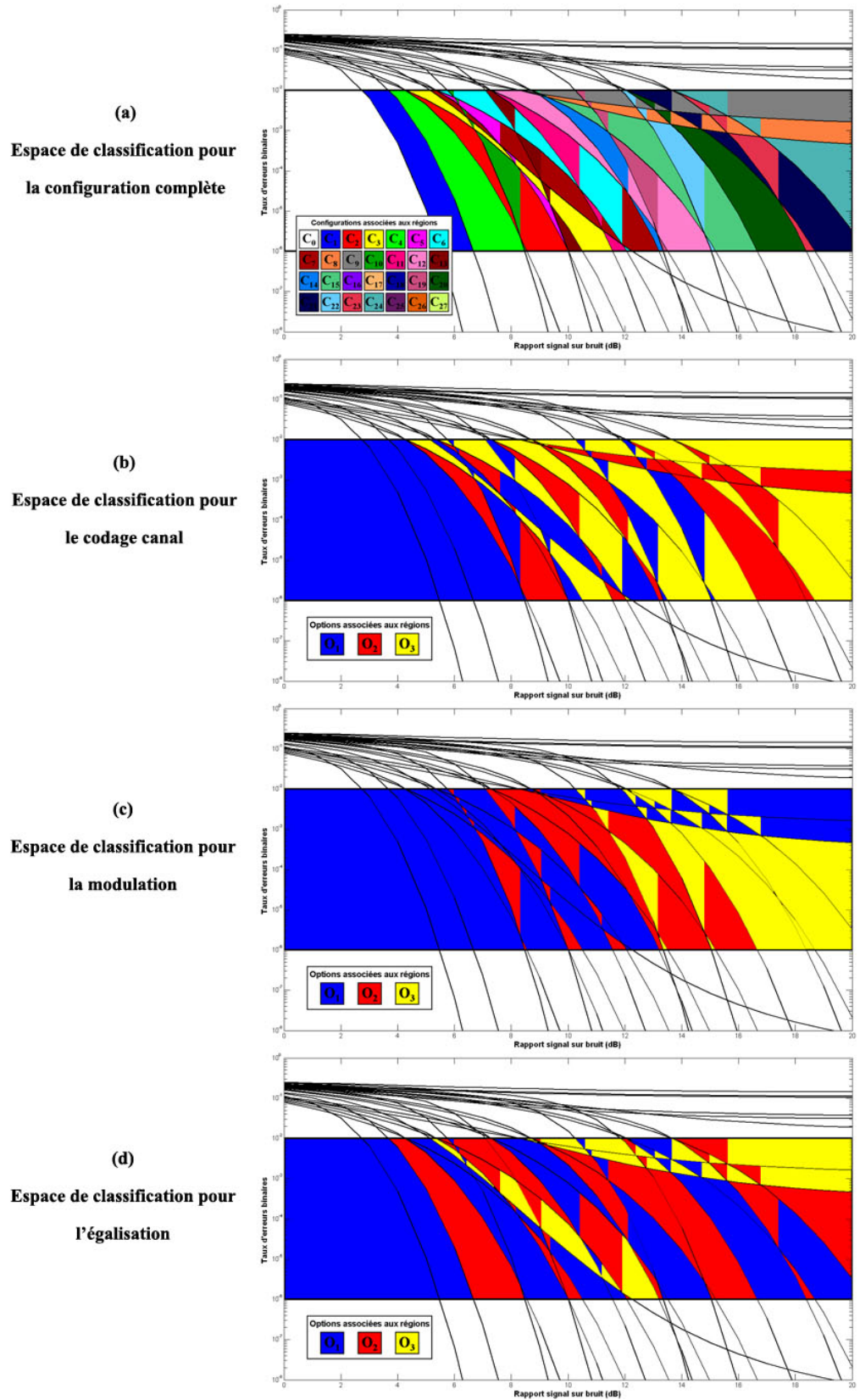


FIGURE 6.4 – Décomposition de l'espace de conception en plusieurs sous-espaces de classification simplifiés

- Les surfaces de décision sont plus faciles à identifier (car moins nombreuses) donc les systèmes prédictifs sont susceptibles de converger plus rapidement vers des prédictions correctes voire optimales.
- Le module CAP n'est pas soumis à une explosion du nombre de classes lorsque la flexibilité de l'équipement augmente. L'introduction d'une nouvelle option de configuration implique la création d'une nouvelle classe pour le système responsable du bloc concerné. Si le bloc n'est pas encore considéré, le module CAP initialise un nouveau système prédictif pour tenir compte de ce degré de flexibilité supplémentaire. Pour la solution monolithique, il aurait fallu ajouter autant de classes que de combinaisons possibles avec la nouvelle option. La décomposition par bloc permet donc de réduire les calculs liés à l'espace de sortie des systèmes prédictifs.
- La spécialisation des modules peut aussi simplifier la gestion de l'espace d'entrée. Le module CAP peut sélectionner les variables pertinentes pour chaque bloc et ainsi diminuer le nombre de dimensions lors du calcul de l'activation des règles. Par exemple, la sélectivité fréquentielle du canal est un facteur essentiel pour la sélection de l'égaliseur approprié. Elle a par contre une influence limitée sur le choix de la modulation et du codage.

Cette solution peut donc avoir son intérêt surtout si elle s'accompagne de procédures d'optimisation assurant une bonne gestion de la complexité des bases de connaissances. Le processus d'expérimentation devra être généralisé dans un premier temps afin de permettre au moteur cognitif d'analyser les résultats d'exploration par bloc.

Gestion des paramètres continus

Le nombre de configurations augmente très rapidement dès que l'on considère des paramètres continus (*e.g.* vitesse de modulation, longueur d'un paquet, puissance d'émission, facteur de roll-off des filtres de mise en forme). Les concepts proposés restent valables pour des espaces aussi larges (*e.g.* la robustesse d'une configuration augmente avec la puissance d'émission). Il sera par contre nécessaire de modifier les mécanismes et les schémas de représentation des connaissances pour contrôler la complexité du moteur cognitif. Les graphes de robustesse doivent être généralisés à leur version continue pour représenter les connaissances expertes de façon compacte. Les règles identifiées ne peuvent plus continuer à mémoriser l'échelle de performance complète. Elles peuvent se contenter de retenir seulement les relations d'ordre impliquées dans la structuration de la région visitée. A titre d'exemple, les règles associées à un faible SNR n'accordent aucune importance aux rangs élevés de l'échelle de performance. Le processus d'expérimentation peut alors reconstituer l'échelle de performance complète en analysant les règles voisines et leur échelle locale afin d'identifier les configurations intéressantes à explorer (*cf.* sec. 6.1.2).

Les algorithmes génétiques présentent des avantages incontestables pour l'exploration de larges espaces de recherche. Notre moteur cognitif peut profiter de cette méthode d'optimisation à deux niveaux :

- Pour le processus de décision, le formalisme de prise de décision multi-critères peut laisser place à un algorithme génétique recherchant la configuration la plus adaptée parmi l'ensemble des configurations compatibles. L'évaluation d'une configuration repose alors sur les prédictions des modules DSS et CAP pour les contraintes de performance et sur les fonctions objectif du système de notation pour les objectifs secondaires.
- Le processus d'expérimentation peut employer un algorithme génétique pour identifier les configurations les plus attrayantes pour l'exploration. Il cherchera à minimiser les risques d'exploration (configuration relativement proche du HCR prédit), à caractériser rapidement

la région considérée (possibilité d'inférer sur les performances de plusieurs configurations grâce au graphe de robustesse) et à améliorer les objectifs opérationnels par rapport à la meilleure configuration connue. Le moteur cognitif peut alors profiter du partitionnement local réalisé par le module DSS pour stocker une population initiale de chromosomes formant la mémoire d'exploration locale.

Le moteur cognitif hériterait alors des avantages de l'approche exploratoire tout en s'affranchissant d'une idéalisation mathématique grâce au développement du modèle prédictif. Ces avantages sont de deux natures. Premièrement, les algorithmes génétiques travaillent avec une représentation des paramètres modifiables. L'exploration est simulée à l'aide des fonctions d'évaluation (expérience de conception virtuelle) donc elle ne perturbe pas la communication en cours. Deuxièmement, il est possible de distribuer la charge de calcul entre plusieurs radios afin de réduire la complexité individuelle de chacune d'elle et d'accélérer la convergence de l'algorithme.

Développement d'une stratégie de communication adéquate

Notre moteur cognitif sans modification n'est pas adapté à la gestion des paramètres continus. Nous noterons cependant qu'il n'est pas toujours intéressant de chercher à optimiser tous les paramètres d'émission/réception du lien radio. Le moteur cognitif serait plus efficace s'il s'appuyait sur les techniques adaptatives existantes (*e.g.* contrôle de puissance, adaptation de modulation et de codage, égalisation adaptative) pour construire une stratégie de communication adéquate face à son contexte opérationnel. Rondeau avait évoqué cette éventualité comme une perspective pour sa thèse [19]. Il avait alors souligné le problème d'établir des fonctions d'évaluation pour guider le processus évolutionnaire. Notre moteur cognitif n'a pas ce problème grâce à ses capacités de modélisation boîte noire. Il sera même avantage dans ce contexte limitant la taille de l'espace de conception.

Partage des connaissances et gestion avancée de la phase d'exploration

Le moteur cognitif a besoin d'un plus grand nombre de tests pour optimiser son comportement sur un large espace de conception. Sa progression vers le comportement optimal est plus lente donc il est nécessaire de mettre en place des mécanismes de désapprentissage plus efficaces. Ce problème peut trouver une solution dans le partage des connaissances entre les radios. Lors de leurs interactions, les radios peuvent échanger leurs règles de conception locales afin de synthétiser le meilleur des connaissances partagées (analyse comparée). Elles peuvent aussi participer à une structuration commune de l'espace de conception qui serait intégrée et relayée par les stations de base. Dans ce scénario, chaque radio remonte ses résultats d'expérimentation vers l'infrastructure de support. Les stations de base profitent alors de leur vision globale pour capitaliser sur toutes les expériences réalisées et identifier les relations d'ordre locales. Elles communiquent ensuite les graphes de robustesse identifiés afin de guider les futures expérimentations et ainsi augmenter progressivement l'efficacité globale du réseau.

Le partage d'informations et de connaissances entre les radios est une solution efficace au problème d'exploration pour de larges espaces de conception. Deux autres pistes sont possibles.

La première piste consiste à multiplier les tentatives d'exploration d'une radio en minimisant leur impact sur l'expérience de l'utilisateur. Le moteur cognitif peut notamment étaler son expérience de conception dans le temps comme nous l'avons suggéré à la section 6.2.2. Il peut aussi s'entraîner lorsque l'équipement est en veille si celui-ci dispose de suffisamment d'énergie et si l'environnement spectral n'est pas trop chargé. Il lance alors une communication test entre la radio supervisée et la station de base la plus proche. Il en profite pour déclencher une phase

d'exploration sans se préoccuper autant des erreurs de décision. Ce scénario d'expérimentation intensive est notamment possible pour les situations fréquemment rencontrées (*e.g.* à domicile, sur le lieu de travail).

La seconde piste passe par la mise au point de connaissances expertes permettant de guider efficacement l'exploration de l'espace de conception. Les travaux de Weingart *et al.* sont particulièrement intéressants dans cette direction (*cf.* sec. 2.4.3).

6.4 Application des concepts à l'accès opportuniste au spectre

Une radio opportuniste passe d'une bande fréquentielle à une autre en fonction de l'activité spectrale autour d'elle. Elle surveille le spectre radio pour réagir à l'apparition d'utilisateurs prioritaires et elle identifie les opportunités spectrales pour optimiser ses objectifs opérationnels. Nos travaux présentent un intérêt pour la prise de décision autonome dans ce contexte.

Notre moteur cognitif peut sélectionner une bande fréquentielle appropriée en analysant ses cartes de performance à la recherche de la meilleure opportunité spectrale. Ces cartes sont caractéristiques de l'interaction entre la radio et son canal de communication. Elles sont construites à partir de statistiques normalisées reliant les caractéristiques du signal de communication avec celles du canal. Le SINR correspond par exemple au rapport entre la puissance d'émission du signal et le niveau d'interférences plus bruit du canal. La bande de cohérence du canal est comparée à la bande passante du signal avant d'être présentée en entrée des systèmes connexionnistes. La fréquence Doppler normalisée est calculée en multipliant la fréquence Doppler du canal avec le temps d'échantillonnage du signal.

Le moteur cognitif peut donc simuler une augmentation de la puissance d'émission en se déplaçant le long de l'axe du SINR dans ses cartes de performance. Il peut généraliser cette simulation à l'aide d'un algorithme de recherche explorant les cartes jusqu'à identifier la meilleure opportunité spectrale. Il sélectionne alors la bande fréquentielle la plus intéressante et il adapte les paramètres d'émission/réception en s'assurant de :

- vérifier les contraintes de régulation,
- satisfaire les exigences de qualité de service,
- optimiser les objectifs secondaires.

Le moteur cognitif peut limiter l'espace de recherche de l'algorithme d'optimisation en pré-traitant les mesures sur le spectre radio afin de sélectionner les opportunités spectrales les plus prometteuses (peu d'interférences, bande large, régulation favorable). Chaque opportunité est caractérisée par sa fréquence centrale f_i , sa largeur de bande B_i , sa quantité d'interférences I_i et son niveau de bruit N_i . Le moteur cognitif analyse les opportunités disponibles en recherchant les paramètres d'émission/réception permettant de les exploiter au mieux. Il peut alors procéder de la manière suivante pour chaque opportunité mise en valeur :

1. Il calcule la puissance maximale tolérable en comparant I_i avec les informations récoltées sur le cadre de régulation. Il en déduit le SINR atteignable.
2. Il consulte son modèle prédictif afin d'identifier l'ensemble des configurations compatibles avec les exigences de QoS.
3. Il recherche la configuration la plus satisfaisante en s'assurant d'écarter celles de bande passante supérieure à B_i (contrainte sur la modulation, la vitesse symbole et le filtrage de mise en forme). Son exploration est guidée par une politique d'optimisation tenant compte des préférences de l'utilisateur (*e.g.* augmenter le débit, préserver l'énergie de la batterie) et de l'opérateur (*e.g.* maximiser la capacité du réseau, assurer une certaine équité

entre les utilisateurs). Les solutions potentielles sont évaluées à l'aide de fonctions objectif paramétrées traduisant la satisfaction de la configuration vis-à-vis des objectifs secondaires considérés. L'algorithme de recherche multiplie les évaluations et les comparaisons jusqu'à identifier une solution particulièrement adaptée.

4. Il parcourt les cartes de performance une seconde fois afin d'estimer la puissance minimum nécessaire pour maintenir la compatibilité de la configuration choisie avec les contraintes de performance. Il simule la diminution de la puissance d'émission en se déplaçant vers les SINR plus faibles. La puissance minimum est obtenue lorsque la configuration choisie correspond au rang de plus haute compatibilité.
5. Il évalue l'opportunité à nouveau pour tenir compte de la diminution éventuelle de la puissance d'émission.

Le moteur cognitif peut ensuite prendre sa décision finale et sélectionner l'opportunité permettant d'optimiser au mieux les objectifs opérationnels. Il garde aussi en mémoire les autres opportunités afin de pouvoir réagir rapidement si le lien radio est dégradé (*e.g.* apparition d'un utilisateur prioritaire, augmentation du niveau d'interférences).

Conclusion et perspectives

Sommaire

7.1	Bilan des travaux réalisés	153
7.2	Contributions	156

7.1 Bilan des travaux réalisés

La radio cognitive apparaît comme une solution naturelle aux problèmes d'échelle et de complexité résultant de la grande popularité des communications sans fil et de l'évolution des technologies radio. Ce nouveau paradigme est directement lié au développement de l'intelligence embarquée, objet de ce travail de thèse.

Dans ce mémoire de thèse, nous avons détaillé la conception d'un moteur cognitif contrôlant l'adaptation d'une radio à son contexte opérationnel. La solution proposée suit une approche originale basée sur une modélisation efficace du problème de conception à la volée. Elle s'inspire de la méthodologie de conception traditionnelle (*cf.* Fig. 1.1) et elle s'appuie sur des techniques d'intelligence artificielle particulièrement adaptées pour la prise de décision autonome. Le comportement du moteur cognitif a été analysé en détail sur deux études de cas pour démontrer la pertinence des mécanismes développés.

Dans le chapitre 2, nous avons défini la notion d'intelligence pour une radio et nous avons étudié les solutions proposées pour la mettre en place. Nous avons ensuite réalisé une analyse critique des approches cognitives existantes afin d'établir un cahier des charges pour la conception de notre moteur cognitif. Nous le rappelons ici brièvement :

1. Le moteur cognitif doit s'affranchir de toute idéalisation mathématique en s'appuyant sur des capacités de prédiction.
2. Il doit pouvoir s'adapter à un nouveau problème en recherchant la solution optimale avec efficacité.
3. Il doit apprendre de ses expériences pour améliorer ses performances face aux problèmes fréquemment rencontrés.
4. Il doit faire évoluer son modèle prédictif au rythme des informations récoltées afin d'augmenter la fiabilité de ses décisions dans le temps (apprentissage incrémental).
5. Il doit savoir exploiter des connaissances expertes pour améliorer la fiabilité de ses mécanismes, augmenter leur rapidité et/ou réduire leur complexité.

6. Il doit maîtriser la complexité de ses processus afin de tenir compte des contraintes d'énergie et de prise de décision temps-réel.
7. Il doit s'appuyer sur des mécanismes suffisamment génériques pour être opérationnel sur de nombreux problèmes de reconfiguration dynamique.

Nous avons répondu à l'ensemble de ces objectifs.

Le moteur cognitif a été conçu à partir d'un modèle original issu de trois observations clés. Nous les résumons ci-dessous en indiquant leur influence sur le développement des mécanismes proposés.

Premièrement, nous avons remarqué que la radio était gouvernée par des objectifs et des contraintes de différente nature. Nous les avons alors hiérarchisés en trois niveaux en fonction de leur impact sur la prise de décision. Le moteur cognitif reproduit cette hiérarchie dans sa manière de résoudre un problème de conception. Dans un premier temps, il profite des capacités du moteur d'inférence pour tenir compte des contraintes de faisabilité imposées par le cadre de régulation et la flexibilité limitée des plateformes matérielles (modélisation experte du contexte opérationnel). Il s'appuie ensuite sur un modèle prédictif pour estimer l'impact du canal radio sur la réalisation des contraintes de performances (processus d'analyse prédictive). Il évalue enfin les préférences d'optimisation afin de rechercher la configuration la plus adaptée parmi les configurations compatibles avec le service requis (processus de décision).

Deuxièmement, nous nous sommes inspirés de la philosophie du soft computing en adoptant une démarche plus qualitative que quantitative afin de proposer un moteur cognitif efficace et robuste. Nous avons alors choisi de raisonner en termes de performances relatives plutôt qu'absolues. Cette démarche est à l'origine de la définition des échelles de performance et d'optimalité. Elle a présenté quatre avantages :

- Elle a simplifié la phase d'analyse prédictive en réduisant le nombre d'informations à mémoriser.
- Elle a introduit une certaine tolérance aux erreurs de prédiction.
- Elle a facilité la conception du processus d'expérimentation (le moteur cognitif navigue le long de l'échelle de performance locale pour réagir aux retours de l'environnement).
- Elle nous a permis d'identifier des connaissances expertes répondant à l'*objectif 5* (le moteur cognitif explore son espace de conception plus efficacement s'il dispose d'un graphe de robustesse particulièrement instructif).

Troisièmement, nous avons traité la tâche de prédiction des performances comme un problème de classification. Nous sommes alors parvenus à donner une interprétation claire de l'espace de conception et des décisions prises par notre moteur cognitif. Cette compréhension nous a permis de mieux orienter nos choix méthodologiques. Elle est aussi à l'origine des mécanismes avancés que nous proposons comme perspectives à ces travaux.

Pour implémenter le modèle, nous avons sélectionné des techniques d'intelligence artificielle affichant des propriétés excellentes dans un contexte de radio cognitive. Ces techniques nous ont permis notamment de répondre aux *objectifs 1, 3 et 4*. Les systèmes connexionnistes évolutifs permettent au moteur cognitif de mémoriser et d'optimiser le modèle prédictif garant de la qualité des décisions prises. Les automates d'apprentissage contrôlent le dilemme exploration/exploitation localement afin de limiter les risques d'expérimentation en fonction de l'expérience du moteur cognitif.

Nous avons proposé une architecture pour structurer les opérations de modélisation, de raisonnement et d'apprentissage. Nous avons décrit les mécanismes cognitifs au chapitre 4 pour un espace de conception totalement ordonné. Nous les avons ensuite généralisés au chapitre 5 pour

traiter le cas plus complexe d'un espace de conception partiellement ordonné. Le moteur cognitif s'appuie sur quatre processus coordonnés par un algorithme de supervision :

- Le processus d'analyse prédictive a deux rôles. Premièrement, il parcourt les bases de connaissances évolutives afin d'évaluer l'adéquation des configurations en fonction de la difficulté du problème (conditions de propagation, exigences de QoS). Deuxièmement, il analyse la stratégie locale d'exploration afin d'estimer l'expérience du moteur cognitif sur la région de l'espace de conception visitée.
- Le processus de décision est activé avec une grande probabilité lorsque la région semble bien caractérisée au regard de la stratégie d'exploration (exploitation des connaissances). Le moteur cognitif sélectionne alors l'alternative la plus satisfaisante parmi les configurations validées par le modèle prédictif.
- Le processus d'expérimentation est déclenché si le moteur cognitif estime que ses connaissances ne sont pas optimales (i.e. la probabilité locale d'exploration est élevée) ou s'il doit réagir à une erreur de prédiction (exploration de l'espace de conception). Le moteur cognitif s'appuie alors sur l'échelle de performance identifiée et sur le graphe de robustesse embarqué pour tenter de se rapprocher du comportement optimal tout en minimisant les erreurs de décision. Cette démarche exploratoire permet au moteur cognitif de s'adapter à toute nouvelle situation (*objectif 2*).
- Le processus d'apprentissage assimile les informations déduites de l'expérimentation afin d'augmenter la fiabilité et la qualité du modèle prédictif. Il met à jour les bases de connaissances entretenues par les systèmes connexionnistes évolutifs et il actualise la stratégie stochastique d'exploration.

Le moteur cognitif a été testé sur deux problèmes de conception dynamique du lien radio. Nous avons proposé un ensemble d'indicateurs de performance afin d'analyser son comportement en détail. Les résultats obtenus nous ont permis de valider l'approche adoptée dans ce travail de thèse. Le moteur cognitif parvient à trouver la solution optimale pour la grande majorité des problèmes traités (95% en moyenne) et il devient de plus en plus efficace dans sa recherche de la solution adaptée. Il perfectionne son expertise sur l'espace de conception en accumulant de l'expérience et il apprend à gérer ses phases d'exploration afin de limiter les risques d'expérimentation. Il peut aussi profiter de connaissances expertes pour améliorer ses performances tout en étant capable de s'en affranchir par apprentissage. La radio dotée de notre moteur cognitif est capable de garantir la qualité de service requise à l'utilisateur en optimisant la gestion des ressources disponibles. Elle est capable également de moduler les objectifs de conception en fonction de son contexte opérationnel.

Dans le chapitre 6, nous avons présenté les mécanismes avancés que nous avons identifiés pour répondre aux *objectifs 6 et 7*. Nous avons proposé de nombreuses améliorations pour réduire la complexité des mécanismes cognitifs et nous avons suggéré plusieurs pistes pour traiter le cas d'un environnement variable dans le temps et d'un espace de conception plus complexe (multiplication des contraintes de performance, augmentation de la flexibilité des équipements radio). Les perspectives envisagées n'ont pas encore été validées expérimentalement. Nous avons amorcé la conception et le développement d'une plateforme de démonstration basée sur GNU Radio [38, 150] pour que ces travaux soient poursuivis.

Nos mécanismes sont suffisamment génériques pour être adaptés à d'autres applications de la radio cognitive comme l'accès opportuniste au spectre. L'approche proposée peut également être exportée à des couches plus élevées de la pile protocolaire. Nous espérons de ce fait que notre solution saura inspirer d'autres travaux poussant le concept de radio cognitive à un niveau encore plus élevé et plus opérationnel.

7.2 Contributions

La table 7.1 résume les contributions des travaux de recherche réalisés dans le cadre de cette thèse. Ces travaux ont été valorisés au travers de 4 communications et d'un article de revue. De nombreux résultats obtenus tardivement restent encore à être publiés. En parallèle de la thèse, j'ai participé au projet IST ORACLE [151] sur la prise de décision autonome dans un contexte d'accès opportuniste au spectre. Ma contribution s'est concrétisée par deux rapports collaboratifs référencés ci-dessous avec les papiers publiés.

Contributions conceptuelles
Analyse des objectifs de conception en fonction de leur impact sur la prise de décision.
Modélisation originale du problème de prédiction des performances représenté comme un problème de classification. Le modèle découle de l'observation graphique de l'espace de conception. Il a donc l'avantage d'être facilement interprétable.
Identification de connaissances expertes permettant d'assister le moteur cognitif dans ses prises de décision. Ces connaissances introduisent des relations d'ordre entre les alternatives disponibles. Elles sont imprécises donc robustes et faciles à obtenir (au préalable ou en ligne). Elles sont codées efficacement sous la forme d'un graphe acyclique orienté directement exploitable par le moteur cognitif.
Conception d'un processus d'expérimentation embarqué. Le moteur cognitif s'appuie sur ses capacités de prédiction et sur des connaissances expertes pour structurer son exploration de l'espace de conception et minimiser les risques de mauvaises décisions.
Gestion innovante du dilemme exploration/exploitation. Le moteur cognitif apprend à contrôler ses expérimentations en fonction de la confiance qu'il porte dans ses connaissances.
Mise au point d'un processus d'apprentissage incrémental pour l'acquisition automatique des connaissances.
Réflexions sur les conditions d'application du moteur cognitif à des problèmes plus complexes et plus réalistes.
Contributions méthodologiques
Classification des approches cognitives existantes en trois courants dominants.
Organisation des mécanismes cognitifs selon une architecture structurée, modulaire et évolutive.
Sélection de techniques d'intelligence artificielle appropriées. Ces techniques, alors absentes de l'état de l'art sur la radio cognitive, s'avèrent être particulièrement adaptées à la prise de décision autonome.
Mise en place d'un cadre évaluation pour expliquer et valider le comportement du moteur cognitif.
Contributions techniques
Implémentation effective du moteur cognitif sur un simulateur C++.
Caractérisation complète de la solution, ses performances et ses limites. Les résultats présentés confirment la pertinence des mécanismes cognitifs proposés.

TABLE 7.1 – Contributions des travaux de recherche réalisés dans le cadre de la thèse

Article de revue internationale

N. Colson, A. Kountouris, A. Wautier, L. Husson, "A Generic Cognitive Framework for Supervising the Radio Dynamic Reconfiguration. An AI approach based on design problem classification", Annals of Telecommunications, special issue on Cognitive Radio, to be published in May 2009

Communications internationales avec actes

N. Colson, A. Kountouris, A. Wautier, L. Husson, "Autonomous Decision Making Process Supporting Cognitive Waveform Design", Proc. PIMRC08, Cannes, France, Sept. 2008

N. Colson, A. Kountouris, A. Wautier, L. Husson, "Autonomous Decision Making Process for Dynamic Reconfiguration of Cognitive Radios", Proc. ICCCN08, Virgin Island, USA, Aug. 2008

N. Colson, A. Kountouris, A. Wautier, L. Husson, "Cognitive Decision Making Process Supervising the Radio Dynamic Reconfiguration", Proc. CROWNCOM08, Singapore, May 2008

Participation au projet IST européen ORACLE

B. Bochow, S. Chantaraskul, N. Colson, A. Kountouris, K. Moessner, S. Thilakawardana, J. Tiemann, "Definition of context filtering mechanisms and policy framework", ORACLE WP4, Deliverable D4.2, May 2007

B. Bochow, N. Colson, L. Huang, A. Kountouris, K. Moessner, S. Thilakawardana, M. Smirnov, "Draft OR Policy Framework", ORACLE WP4, Deliverable D4.1, Nov. 2006

Communication nationale sans acte

N. Colson, A. Kountouris, "Une application de la radio intelligente : Adaptation dynamique de la complexité algorithmique", Journées Scientifiques du CNFRS intitulée "Vers des radiocommunications reconfigurables et cognitives", Paris, France, Mars 2006

Rapports internes

N. Colson, A. Kountouris "Wireless Radio and the Internet of Things : a case for Cognitive Radio", Tech. Report, France Telecom R&D, Novembre 2007

N. Colson "Étude bibliographique : Réseaux de neurones", Tech. Report, France Telecom R&D, Novembre 2006

N. Colson "Étude bibliographique : Techniques de prise de décision", Tech. Report, France Telecom R&D, Octobre 2006

N. Colson "Étude bibliographique : Canal radio", Tech. Report, France Telecom R&D, Octobre 2006

N. Colson "Étude bibliographique : Logique floue et théorie des possibilités", Tech. Report, France Telecom R&D, Septembre 2006

N. Colson "Étude bibliographique : Systèmes experts et systèmes experts flous", Tech. Report, France Telecom R&D, Août 2006

A

Modélisation analytique du système de communication reconfigurable

Le simulateur s'appuie sur des expressions analytiques pour calculer le BER résiduel en fonction de la configuration choisie et de l'état du canal simulé. Une configuration est représentée par les paramètres de ses blocs reconfigurables (*e.g.* ordre de modulation, longueur des mots de codes, type d'égaliseur). Un canal radio est caractérisé par une valeur de SNR et par un modèle de propagation donnés. Notre moteur cognitif est testé selon une approche statique dans le cadre de cette thèse. L'implémentation actuelle du simulateur ne tient donc pas compte du caractère dynamique des canaux radiomobiles.

Le tableau A.1 résume les notations utilisées dans cette annexe. Le simulateur procède en trois étapes pour calculer les performances d'une configuration donnée :

- Calcul de la dégradation du SNR ($0 < \varepsilon \leq 1$) résultant des interférences inter-symbole plus ou moins bien compensées par l'égaliseur éventuellement activé
- Calcul du BER brut en sortie du modulateur
- Calcul du BER résiduel après décodage d'erreurs

Notation	Paramètre
SNR	Rapport signal sur bruit
ε	Dégradation du SNR (facteur multiplicatif)
M	Ordre de modulation
n	Longueur d'un mot de code
t	Pouvoir correcteur du code
L	Longueur de l'égaliseur linéaire

TABLE A.1 – Notations

Un canal AWGN n'introduit pas d'interférences inter-symbole donc $\varepsilon = 1$. Avec un modèle sélectif de canal, le signal de communication est soumis au phénomène de multi-trajets créant des interférences plus ou moins destructrices au niveau de l'antenne de réception. Ces interférences peuvent être compensées par un égaliseur (au moins partiellement). La dégradation du SNR dépend alors de la présence/absence de l'égaliseur et de la technique utilisée :

- L'égaliseur MLSE est un égaliseur particulièrement performant [110] donc nous avons fait l'approximation d'interférences parfaitement compensées ($\varepsilon = 1$).

- La dégradation du SNR pour un égaliseur linéaire peut se calculer à partir d'expressions analytiques disponibles dans [152]. Elle dépend de la longueur de l'égaliseur $\rightarrow \varepsilon = f(L)$.
- Les mêmes expressions peuvent être utilisées pour calculer la dégradation du SNR pour un récepteur sans égaliseur en prenant $L = 1 \rightarrow \varepsilon = f(1)$.

La valeur du SNR alimente les expressions analytiques fournissant le BER brut en sortie du modulateur. Ces expressions sont disponibles dans la plupart des livres classiques sur les communications radionumériques (*e.g.* [4]). Pour les modulations M-PSK, le BER brut est obtenu à partir de l'équation A.1 pour $M = 2$ et de l'équation A.2 pour les modulations d'ordre supérieur ($M > 2$).

$$BER_{\text{Brut}} = \frac{1}{2} \operatorname{erfc} \left(\sqrt{\varepsilon SNR} \right) \quad (\text{A.1})$$

$$BER_{\text{Brut}} = \frac{1}{\log_2(M)} \operatorname{erfc} \sqrt{\varepsilon SNR \sin \left(\frac{\pi}{M} \right)} \quad (\text{A.2})$$

Le BER résiduel après correction d'erreurs peut être calculé à partir de l'équation A.2.

$$BER_{\text{Résiduel}} \simeq \sum_{j=t+1}^n \frac{j+t}{n} \binom{n}{j} (BER_{\text{Brut}})^j (1 - BER_{\text{Brut}})^{n-j} \quad (\text{A.3})$$


B

Compléments d'information sur les mécanismes cognitifs

B.1 Processus d'expérimentation généralisé

Exemple d'application de la règle déterministe d'exploration offensive

La figure B.1 montre un exemple d'application de la règle déterministe d'exploration offensive. L'exemple proposé s'appuie sur l'espace de conception à 10 configurations présenté à la figure 5.2 (les liens orientés précisent les relations d'ordre entre les alternatives considérées). Dans cet exemple, le processus d'expérimentation applique la règle indéfiniment pour caractériser complètement l'espace de conception local. L'expérimentation n'est pas aussi poussée en pratique afin de tenir compte de l'impact des erreurs de décision sur l'expérience de l'utilisateur. Pour des raisons de clarté, nous n'encombrons pas l'exemple des informations de mise à jour des connaissances (échelle de performance et HCR identifiés).



Etat initial	C _a	C _b	C _c	C _d	C _e	C _f	C _g	C _h	C _i	C _j
Test du HCR	C _a	C _b	C _c	C _d	C _e	C _f	C _g	C _h	C _i	C _j
Exploration n°1	C _a	C _b	C _c	C _d	C _e	C _f	C _g	C _h	C _i	C _j
Exploration n°2	C _a	C _b	C _c	C _d	C _e	C _f	C _g	C _h	C _i	C _j
Exploration n°3	C _a	C _b	C _c	C _d	C _e	C _f	C _g	C _h	C _i	C _j
Exploration n°4	C _a	C _b	C _c	C _d	C _e	C _f	C _g	C _h	C _i	C _j

FIGURE B.1 – Application de la règle déterministe d'exploration offensive

Le tableau d'exploration est initialisé à l'état *I* (police normale) pour toutes les configurations disponibles. Nous faisons l'hypothèse que l'échelle de performance part de *C_a* pour se terminer à *C_j* et que *C_d* est la configuration compatible la moins robuste (HCR = 4). De plus, la région associée au problème est supposée très mal caractérisée (probabilité d'exploration locale élevée, poids d'exploration minimum pour toutes les configurations). Le moteur cognitif cherche donc à mettre ses connaissances à l'épreuve en déclenchant le processus d'expérimentation après avoir

testé C_d . Ses connaissances ne sont pas erronées puisque C_d vérifie la contrainte. Elles peuvent cependant être sous-optimales d'où le besoin d'explorer. Le moteur cognitif commence par mettre le tableau d'exploration à jour en passant C_a , C_b et C_d à l'état *CS* (en gras)⁴⁰. Le processus d'expérimentation repère ensuite toutes les opportunités de progression (*i.e.* C_e , C_f , C_g , C_h , C_i et C_j) puis il exclut C_g , C_i et C_j moins robustes respectivement que C_e , C_f et C_e à nouveau. Les poids d'exploration sont supposés égaux pour les trois configurations restantes (*i.e.* C_e , C_f , C_h). Le processus d'expérimentation décide donc de tester C_e , l'opportunité de plus faible rang. La configuration C_e n'est pas assez performante pour atteindre les exigences de QoS. Le moteur cognitif actualise le tableau d'exploration en éliminant C_e , C_g et C_j des opportunités d'exploration (passage à l'état *CV* en italique). Il applique ensuite la même logique pour prendre ses prochaines décisions (explorations 2 à 4).

Minimisation des poids d'exploration lors de la prise de décision statistique sur l'arrêt de l'expérimentation

Considérons trois opportunités d'exploration de poids respectifs $w_1 = 6$, $w_2 = 5$ et $w_3 = 3$. L'algorithme calcule la probabilité $p_1 = 0,16$ pour la configuration de plus bas rang dans l'échelle de performance. Le moteur cognitif prend une décision stochastique en tirant un nombre aléatoire p . Si $p < p_1$, le processus d'expérimentation sélectionne la configuration associée à w_1 . Il calcule $p_2 = 0,21$ dans le cas contraire. Il sélectionne alors la configuration associée à w_2 si $p < p_1 + p_2$ et la configuration associée à w_3 sinon. La somme des probabilités est égale à 1 donc $p_3 = 0,63$. Sans la normalisation à 1 du poids minimum, on aurait $p_1 = 0,24$, $p_2 = 0,28$ et $p_3 = 0,48$. La configuration associée à w_3 aurait donc moins de chance d'être sélectionnée.

B.2 Résultats supplémentaires

Première étude de cas

La figure B.2 affiche la carte d'exploration obtenue à différentes étapes de la simulation. Les cartes d'exploration suivent l'évolution de la probabilité moyenne d'exploration. Elles restent relativement homogènes sur l'intégralité de l'espace de conception car nous avons choisi un large rayon de voisinage ($R_V = 0,2$).

Seconde étude de cas

La figure B.3 affiche la carte d'exploration à différentes étapes de la simulation. Nous avons analysé la stabilité exploratoire des régions de l'espace directement dans le corps du texte.

40. La configuration C_c reste à l'état indéterminé puisqu'il n'est pas possible d'inférer sur ses performances à partir de C_d .

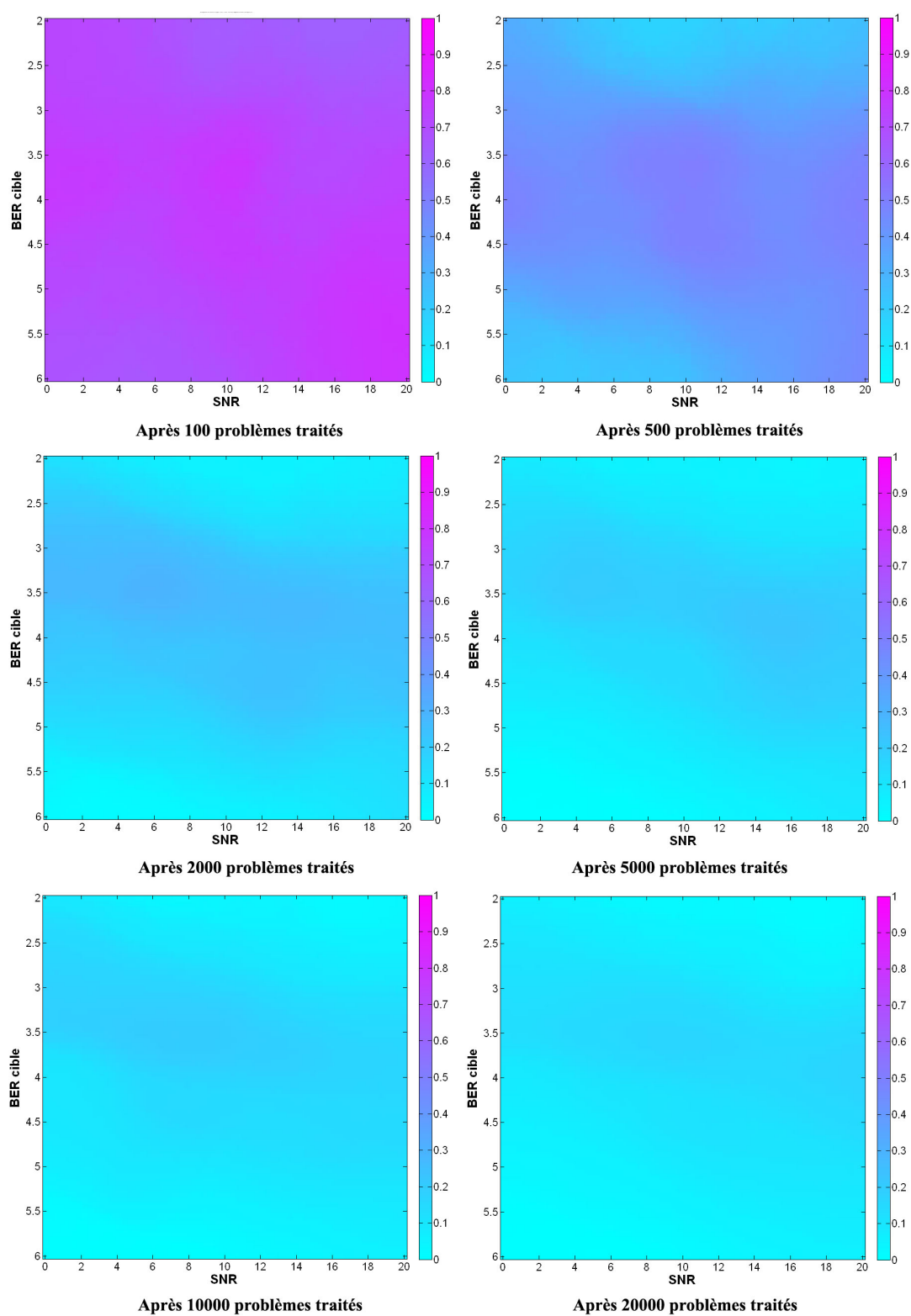


FIGURE B.2 – Evolution de la carte d’exploration ($PSSE_2$) avec l’expérience du moteur cognitif pour la première étude de cas

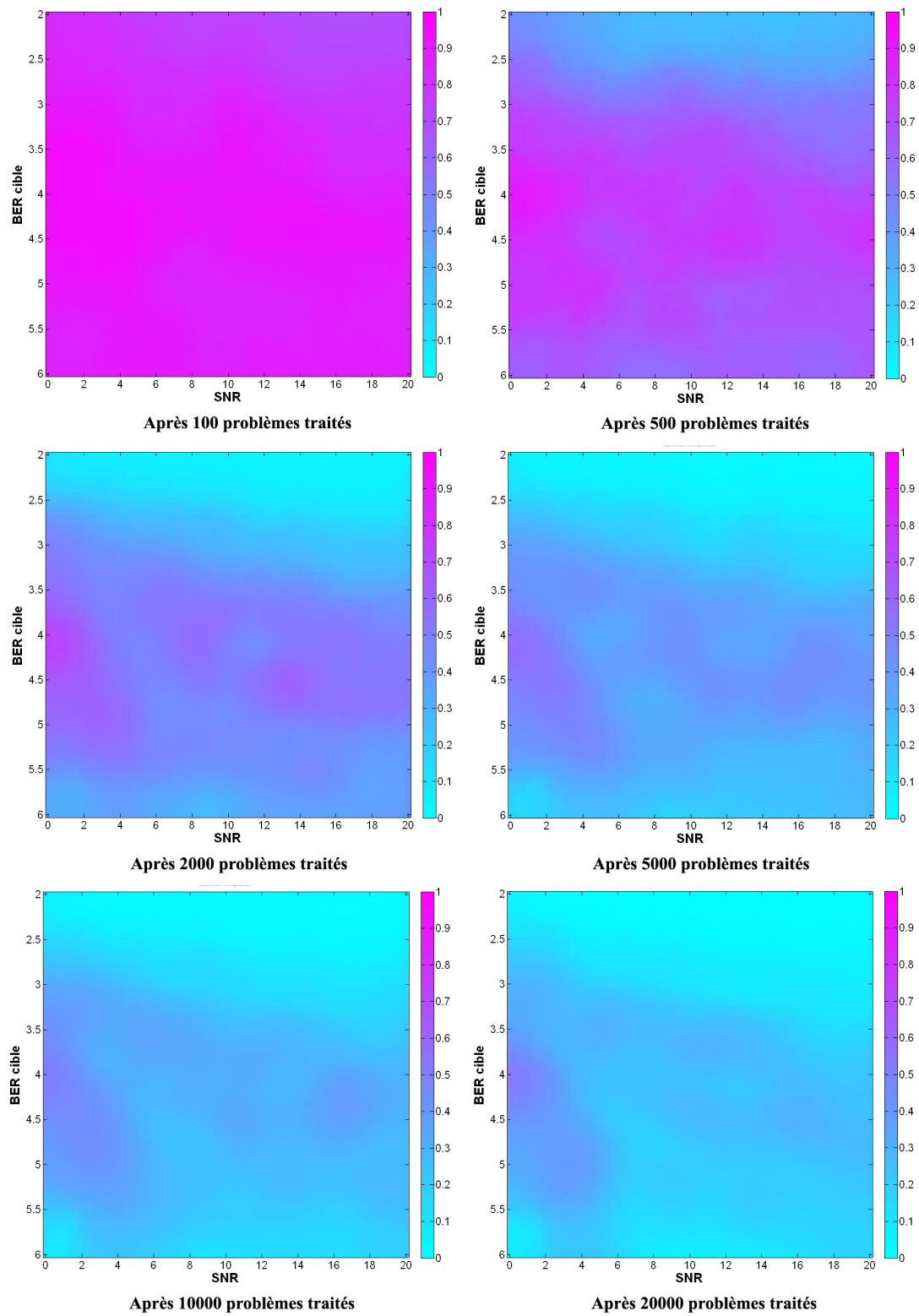


FIGURE B.3 – Evolution de la carte d'exploration ($PSSE_2$) avec l'expérience du moteur cognitif pour la seconde étude de cas

Glossaire

ASIC : Application-Specific Integrated Circuit

AWGN : Additive White Gaussian Noise

BCH : Bose, Ray-Chaudhuri, Hocquenghem

BER : Bit Error Rate

BPSK : Binary Phase Shift Keying

CAP : Configuration Adequacy Predictor

CSM : Context Sensing Manager

CWT : Cognitive Wireless Technology

DARPA : Defense Advanced Research Projects Agency

dB : Décibel

DSS : Design Space Structurer

ECF : Evolving Classification Function

ECMC : Evolving Clustering Method for Classification

EDGE : Enhanced Data Rates for GSM Evolution

EFuNN : Evolving Fuzzy Neural Networks

ESOM : Evolving Self-Organizing Maps

FPGA : Field-Programmable Gate Array

GNU : Gnu's Not Unix

GPRS : General Packet Radio Service

GS : Grading System

GSM : Global System for Mobile Communications

HCR : Highest Compatible Rank

HSDPA : High-Speed Downlink Packet Access

M2M : Machine to Machine

MLSE : Maximum Likelihood Sequence Estimation

NNM : Neural Network Module

OP : Optimization Policy

OS : Optimality Scale

OSSIE : Open Source SCA Implementation : :Embedded

OWL : Web Ontology Language

PEW : Problem Evolution Watcher

PS : Performance Scale

PSK : Phase Shift Keying

QAM : Quadrature Amplitude Modulation

QoS : Quality of Service

QPSK : Quadrature Phase Shift Keying

RALFE : Reason And Learn From Experience

RBA : Radio Behavior Analyzer

RDF : Resource Description Framework

SC : Satisfied Constraint

SCA : Software Communications Architecture

SINR : Signal to Interference-plus-Noise Ratio

SNR : Signal-to-NoiseRatio

TOPSIS : Technique for Order Preference by Similarity to Ideal Solution

VC : Violated Constraint

XG : neXt Generation

XML : eXtensible Markup Language

Références bibliographiques

- [1] A. Molisch, *Wireless Communications*. Wiley-IEEE Press, Sept. 2005.
- [2] B. Sklar, “Rayleigh fading channels in mobile digital communication systems part i : Characterization,” *IEEE Communications Magazine*, pp. 90–100, July 1997.
- [3] —, “Rayleigh fading channels in mobile digital communication systems part ii : Mitigation,” *IEEE Communications Magazine*, pp. 102–109, July 1997.
- [4] J. G. Proakis, *Digital Communications*, 4th ed. New York : McGraw-Hill, 2000.
- [5] COST. 259, “Technical specification group radio access networks, deployment aspects (release 6),” 3rd Generation Partnership Project, Tech. Rep., Dec. 2004.
- [6] C. J. Rieser, T. W. Rondeau, C. Bostian, W. R. Cyre, and T. M. Gallagher, “Cognitive radio engine based on genetic algorithms in a network,” Patent Application, Publication n°US2006 / 0009092 A1, Jan 2006.
- [7] P. Tortelier, “A policy description language for policy-defined cognitive radio,” Orange Labs, Tech. Rep., June 2008.
- [8] J. B. Evans, “Intelligence in the network,” Keynote Speech, International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom), June 2008.
- [9] J. Mitola, “Software radios - survey, critical evaluation and future directions,” in *IEEE National Telesystems Conference*, May 1992, pp. 13/15 – 13/23.
- [10] —, *Software Radio Architecture*. Wiley, Oct. 2000.
- [11] J. H. Reed, *Software Radio : A Modern Approach to Radio Engineering*. Prentice Hall, May 2002.
- [12] International Telecommunication Union (ITU), “The internet of things - executive summary,” Tech. Rep., Nov. 2005.
- [13] Spectrum Policy Task Force, “Sets forth recommendations for spectrum policy reform,” Federal Communications Commission, Tech. Rep., Nov. 2002.
- [14] H. Zheng and C. Peng, “Collaboration and fairness in opportunistic spectrum access,” in *International Conference on Communications (ICC)*, vol. 5, May 2005, pp. 3132–3136.
- [15] J. O. Kephart and D. M. Chess, “The vision of autonomic computing,” *Computer*, vol. 36, no. 1, pp. 41–50, Jan. 2003.
- [16] J. Mitola and G. Maguire, “Cognitive radio : Making software radios more personal,” in *IEEE Personal Communication*, vol. 6, no. 4, Aug. 1999, pp. 13–18.
- [17] J. D. Neel, “Analysis and design of cognitive radio networks and distributed radio resource management algorithms,” Ph.D. dissertation, Virginia Polytechnic Institute and State University, Sep. 2006.

- [18] T. C. Clancy, "Dynamic spectrum access in cognitive radio networks," Ph.D. dissertation, University of Maryland, College Park, April 2006.
- [19] T. W. Rondeau, "Application of artificial intelligence to wireless communications," Ph.D. dissertation, Virginia Polytechnic Institute and State University, Sep. 2007.
- [20] F. K. Jondral, "Cognitive radio : A communications engineering view," *Wireless Communications*, vol. 14, no. 4, pp. 28–33, Aug 2007.
- [21] D. Domenicali, "Design of uwb networks based on the application of the cognitive radio paradigm," Ph.D. dissertation, Sapienza Università Di Roma, Mar 2008.
- [22] D. Maldonado, B. Le, A. Hugine, T. W. Rondeau, and C. W. Bostian, "Cognitive radio applications to dynamic spectrum allocation : a discussion and an illustrative example," in *IEEE Proc. DySPAN*, Nov 2005, pp. 597 – 600.
- [23] Federal Communications Commission, "Notice of proposed rulemaking and order," Tech. Rep. ET Docket n°03-322, Dec. 2003.
- [24] P. Ballon, O. Braet, E. Buracchini, S. Delaere, B. Deschamps, O. Holland, J. M. Temerson, K. Kominaki, K. Moessner, F. Negredo, E. Patouni, O. Simon, M. Stamatelatos, and A. Trogolo, "Business models, user value and roadmaps," European IST Project, End-To-End Reconfigurability, E²R II Deliverable D1.1, Tech. Rep., July 2006.
- [25] B. A. Fette, *Cognitive Radio Technology*. Newnes, Sept. 2006.
- [26] A. Alptekin, J. Bradford, E. Burbidge, D. Chandler, M. Kennett, P. Levine, and S. Weiss, "Cognitive radio technology," QinetiQ, Tech. Rep., Feb. 2007.
- [27] J. Grosspietsch, D. Maldonado, K. Nolan, J. Polson, J. Kearns, S. O'Hara, H. Arslan, S. Seidel, D. Shrum, P. Cook, J. Neel, R. Berezdivin, P. O'Horo, R. Janka, B. Breinig, D. Bengston, C. Sheehe, Z. Kostic, B. Fette, K. Pande, J. Mitola, and M. Kokar, "Cognitive radio definitions and nomenclature," SDR Forum, Tech. Rep., Sept. 2008.
- [28] M. Matinmikko, M. Höyhty, M. Mustonen, H. Sarvanko, A. Hekkala, M. Katz, A. Mämmelä, M. Kiviranta, and A. Kautio, "Cognitive radio : An intelligent wireless communication system," VTT Technical Research Centre of Finland, Tech. Rep., March 2008.
- [29] S. Haykin, "Cognitive radio : Brain-empowered wireless communications," *IEEE Journal on Selected Area in Communications*, vol. 23, no. 2, pp. 201–220, Feb. 2005.
- [30] T. W. Rondeau, C. W. Bostian, D. Maldonado, A. Ferguson, S. Ball, B. Le, and S. Midkiff, "Cognitive radios in public safety and spectrum management," in *Telecommunications Policy and Research Conference*, vol. 33, Sept 2005.
- [31] N. Baldo and M. Zorzi, "Learning and adaptation in cognitive radios using neural networks," in *IEEE Consumer Communications and Networking Conference (CCNC)*, Las Vegas, USA, Jan 2008, pp. 998–1003.
- [32] S. Russell and P. Norvig, *Artificial Intelligence : A Modern Approach*, 2nd ed. Prentice-Hall, 1995.
- [33] C. Clancy, J. Hecker, E. Stuntebeck, and T. O'Shea, "Applications of machine learning to cognitive radio networks," *IEEE Wireless Communications*, vol. 14, no. 4, pp. 47–52, Aug 2007.
- [34] C. J. Rieser, "Biologically inspired cognitive radio engine model utilizing distributed genetic algorithms for secure and robust wireless communications and networking," Ph.D. dissertation, Virginia Polytechnic Institute and State University, Aug. 2004.

-
- [35] J. Mitola, "Cognitive radio : An integrated agent architecture for software defined radio," Ph.D. dissertation, Royal Institute of Technology (KTH), Stockholm, Sweden, May 2000.
 - [36] D. Cabric and R. W. Brodersen, "Physical layer design issues unique to cognitive radio systems," in *16th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, (PIMRC)*, vol. 2, Sept. 2005, pp. 759–763.
 - [37] G. J. Minden, J. B. Evans, L. Searl, D. DePardo, V. Petty, R. Rajbanshi, T. Newman, Q. Chen, F. Weidling, J. Guffey, D. Datla, B. Barker, M. Peck, B. Cordill, A. M. Wyglinski, and A. Agah, "KUAR : A flexible software-defined radio development platform," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Apr. 2007, pp. 428–439.
 - [38] Free Software Foundation, "GNU radio project," <http://www.gnuradio.org>.
 - [39] G. D. Troxel and al., "Adaptive dynamic radio open-source intelligent team (adroit) : Cognitively-controlled collaboration among sdr nodes," in *First IEEE Workshop on Networking Technologies for Software Defined Radio (SDR) Networks*, Reston, VA, USA, Sept. 2006.
 - [40] "Open-source SCA implementation : :embedded," <http://ossie.mprg.org>.
 - [41] E. Stuntebeck, T. O'Shea, J. Hecker, and C. Clancy, "Architecture for an open-source cognitive radio," in *SDR Forum Technical Conference*, Nov. 2006.
 - [42] P. Sutton, L. E. Doyle, and K. E. Nolan, "A reconfigurable platform for cognitive networks," in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, June 2006.
 - [43] K. E. Nolan, P. D. Sutton, and L. E. Doyle, "An encapsulation for reasoning, learning, knowledge representation, and reconfiguration cognitive radio elements," in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (Crown-com)*, June 2006.
 - [44] K. E. Nolan, T. W. Rondeau, P. D. Sutton, C. W. Bostian, and L. E. Doyle, "A framework for implementing cognitive functionality," in *51st SDR Forum General Meeting and Technical Conference*, Orlando, USA, Nov 2006.
 - [45] L. Godard, C. Moy, and J. Palicot, "From a configuration management to a cognitive radio management of sdr systems," in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, June 2006.
 - [46] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks : A survey," *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, Sept. 2006.
 - [47] M. Ghozzi, M. Dohler, F. Marx, and J. Palicot, "Cognitive radio : Methods for detection of free bands," *Elsevier Science Journal, Special Issue on Cognitive Radio*, vol. 7, pp. 794–805, Sept 2006.
 - [48] Z. Tian and G. B. Giannakis, "A wavelet approach to wideband spectrum sensing for cognitive radios," in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, 2006.
 - [49] Y.-C. Liang, "Blind spectrum sensing algorithms for cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 5, pp. 2834–2842, Sept 2008.
 - [50] A. Kountouris, P. Marques, J. Bastos, A. Pereira, A. Gameiro, M. Ghozzi, M. Dohler, P. Jallon, S. Thilakawardana, and P. Ashish, "Sensing algorithms part i," The ORACLE Project - Opportunistic Radio Communications in Unlicensed Environments, Tech. Rep.

- [51] —, “Sensing algorithms part ii,” The ORACLE Project - Opportunistic Radio Communications in Unlicensed Environments, Tech. Rep.
- [52] S. M. Mishra, A. Sahai, and R. W. Brodersen, “Cooperative sensing among cognitive radios,” in *International Conference on Communications (ICC)*, Istanbul, Turkey, June 2006.
- [53] W. Yang, Y. Cai, and Y. Xu, “A fuzzy collaborative spectrum sensing scheme in cognitive radio,” in *International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, Nov 2007, pp. 566–569.
- [54] A. Kountouris, B. Bochow, M. Smirnov, and J. Bastos, “Preliminary specification of the basic protocol architecture,” The ORACLE Project - Opportunistic Radio Communications in Unlicensed Environments, Tech. Rep.
- [55] J. Mitola, “Cognitive radio for flexible mobile multimedia communications,” in *IEEE Mobile Multimedia Conference*, Nov. 1999, pp. 3–10.
- [56] T. A. Weiss and F. Jondral, “Spectrum pooling : an innovative strategy for the enhancement of spectrum efficiency,” *IEEE Communications Magazine*, vol. 42, no. 3, pp. S8–14, March 2004.
- [57] “The end-to-end reconfigurability (E2R) project,” <http://e2r2.motlabs.com/>.
- [58] J. Perez-Romero, O. Salient, R. Agustí, and L. Giupponi, “A novel on-demand cognitive pilot channel enabling dynamic spectrum allocation,” in *International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Apr. 2007, pp. 46–54.
- [59] C. Moy, A. Bisiaux, and S. Paquelet, “An ultra-wide band umbilical cord for cognitive radio systems,” in *International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, vol. 2, Sept. 2005, pp. 775–779.
- [60] W. Xu, P. Kamat, and W. Trappe, “Trieste : A trusted radio infrastructure for enforcing spectrum etiquettes,” in *First IEEE Workshop on Networking Technologies for Software Defined Radio Networks*, Reston, VA, USA, Sept. 2006.
- [61] T. Clancy and N. Goergen, “Security in cognitive radio networks : Threats and mitigation,” in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, May 2008.
- [62] J. L. Burbank, “Security in cognitive radio networks : The required evolution in approaches to wireless network security,” in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, May 2008.
- [63] R. W. Thomas, “Cognitive networks,” Ph.D. dissertation, Virginia Polytechnic Institute and State University, June 2007.
- [64] J. Xie, I. Howitt, and A. Raja, “Cognitive radio resource management using multi-agent systems,” in *IEEE Consumer Communications and Networking Conference (CCNC)*, Jan. 2007, pp. 1123–1127.
- [65] T. Renk, C. Kloeck, D. Burgkhardt, F. K. Jondral, D. Grandblaise, S. Gault, and J. C. Dunat, “Bio-inspired algorithms for dynamic resource allocation in cognitive wireless networks,” in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, Orlando, FL, USA, Aug. 2007, pp. 351–356.
- [66] C. Doerr, D. Grunwald, and D. C. Sicker, “Local independent control of cognitive radio networks,” in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, Singapore, May 2008, pp. 1–9.

-
- [67] J.-P. Briot and Y. Demazeau, *Principes et architecture des systèmes multi-agents*. Hermès, Nov. 2001.
 - [68] D. H. Friend, M. Y. ElNainay, Y. Shi, and A. B. MacKenzie, “Architecture and performance of an island genetic algorithm-based cognitive network,” in *Consumer Communications and Networking Conference (CCNC)*, Baltimore, MD, USA, Jan 2008, pp. 993–997.
 - [69] Y. Collette and P. Siarry, *Optimisation multiobjectif*. Eyrolles, Oct. 2002.
 - [70] T. W. Rondeau, B. Le, D. Maldonado, D. Scaperoth, , and C. W. Bostian, “Cognitive radio formulation and implementation,” in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, Mykonos, Greece, June 2006.
 - [71] L. Bossuet, “Exploration de l’espace de conception des architectures reconfigurables,” Ph.D. dissertation, Université de Bretagne Sud, Sept 2004.
 - [72] M. Gries, “Methods for evaluating covering the design space during early design development,” Electronics Research Laboratory, University of California at Berkeley, Technical Memorandum MO3/32, Aug 2003.
 - [73] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA : Addison-Wesley, 1989.
 - [74] M. M. Kokar, D. Brady, and K. Baclawski, *Cognitive Radio Technology*. Newnes, 2006, ch. Roles of Ontologies in Cognitive Radios, pp. 401–433.
 - [75] R. J. Wellington, *Cognitive Radio Technology*. Newnes, 2006, ch. Cognitive Policy Engines, pp. 185–218.
 - [76] L. Berlemann, S. Mangold, and B. H. Walke, “Policy-based reasoning for spectrum sharing in radio networks,” in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Baltimore, MD, USA, Nov. 2005.
 - [77] B. Bochow, S. Chantaraskul, N. Colson, A. Kountouris, K. Moessner, S. Thilakawardana, and J. Tiemann, “Definition of context filtering mechanisms and policy framework,” The ORACLE Project - Opportunistic Radio Communications in Unlicensed Environments, Tech. Rep.
 - [78] DARPA XG Working Group, “The XG vision. request for comments,” BBN Technologies, Cambridge MA, USA, Tech. Rep. Version 2.0, Jan. 2004.
 - [79] XG Working Group, “XG policy language : Cognitive policy radio language (CoRaL), a language for spectrum policies,” Defense Advanced Research Projects Agency, Tech. Rep. Version 0.1, Apr. 2007.
 - [80] M. McHenry, E. Livsics, T. Nguyen, and N. Majumdar, “XG dynamic spectrum access field test results,” *IEEE Communications Magazine*, vol. 45, no. 6, pp. 51–57, Jun. 2007.
 - [81] D. Wilkins, G. Denker, M.-O. Stehr, D. Elenius, R. Senanayake, and C. Talcott, “Policy-based cognitive radios,” *IEEE Wireless Communications*, vol. 14, no. 4, pp. 41–46, Aug. 2007.
 - [82] C. Chien, M. B. Srivastava, R. Jain, P. Lettieri, V. Aggarwal, and R. Sternowski, “Adaptive radio for multimedia wireless links,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 5, pp. 793–813, May 1999.
 - [83] T. G. Dietterich and P. Langley, *Cognitive Networks*. John Wiley & Sons, Jul 2007, ch. Machine Learning for Cognitive Networks : Technology Assessment and Research Challenges, pp. 97–120.

- [84] T. W. Rondeau, C. J. Rieser, T. M. Gallagher, and C. Bostian, "Online modeling of wireless channels with hidden markov models and channel impulse responses for cognitive radios," in *International Microwave Symposium*, Fort Worth, TX, June 2004.
- [85] C. J. Rieser, T. W. Rondeau, and C. W. Bostian, "Cognitive radio testbed : Further details and testing of a distributed genetic algorithm based cognitive engine for programmable radios," in *IEEE Proc. MILCOM (Military Communications Conference)*, Monterey, CA, 2004, pp. 1437 – 1443.
- [86] T. W. Rondeau, C. J. Rieser, B. Le, and C. W. Bostian, "Cognitive radios with genetic algorithms : Intelligent control of software defined radios," in *SDR Forum Technical Conference*, Phoenix, AZ, 2004, pp. C-3 – C-8.
- [87] T. W. Rondeau and C. W. Bostian, *Cognitive Radio Technology*. Newnes, 2006, ch. Cognitive Techniques : Physical and Link Layers, pp. 219–268.
- [88] K. E. Nolan, T. W. Rondeau, P. Sutton, B. Le, L. E. Doyle, and C. W. Bostian, "Demonstration and analysis of collaboration, co-existence and interoperability of cognitive radio platforms," in *1st IEEE Workshop on Cognitive Radio Networks, CCNC*, Las Vegas, USA, Jan 2007.
- [89] —, "Dynamic spectrum access and coexistence experiences involving two independently developed cognitive radio testbeds," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN'07)*, Dublin, Ireland, Apr 2007, pp. 270 –275.
- [90] B. Le, P. Garcia, Q. Chen, B. Li, F. Ge, M. ElNainay, T. W. Rondeau, and C. W. Bostian, "A public safety cognitive radio node system," in *SDR Forum Technical Conference*, Denver, Colorado, Nov 2007.
- [91] F. Ge, Q. Chen, Y. Wang, T. W. Rondeau, B. Le, and C. W. Bostian, "Cognitive radio : From spectrum sharing to adaptive learning and reconfiguration," in *IEEE Aerospace Conference*, Big Sky, Montana, March 2008.
- [92] T. R. Newman, B. A. Barker, A. M. Wyglinski, A. Agah, J. B. Evans, and G. J. Minden, "Cognitive engine implementation for wireless multicarrier transceivers," *Wireless Communications & Mobile Computing*, vol. 7, no. 9, pp. 1129 – 1142, Nov 2007.
- [93] H. I. Volos, C. I. Phelps, and R. M. Buehrer, "Initial design of a cognitive engine for mimo systems," in *SDR Forum Technical Conference*, Denver, Colorado, Nov 2007.
- [94] K. Moessner, S. Chantaraskul, D. Witaszek, B. Bochow, A. Mauch, and F. Xu, "Or decision making engine definition," The ORACLE Project - Opportunistic Radio Communications in Unlicensed Environments, Tech. Rep.
- [95] A. Aamodt and E. Plaza, "Case-based reasoning : Foundational issues, methodological variations, and system approaches," *Artificial Intelligence Communications*, vol. 7, no. 1, pp. 39–59, 1994.
- [96] S. K. Pal and S. C. K. Shiu, *Foundations of Soft Case-Based Reasoning*. John Wiley & Sons, Online Edition, June 2004.
- [97] C. Alasseur, S. Scalise, L. Husson, and H. Ernst, "A novel approach to model the land mobile satellite channel through reversible jump markov chain monte carlo technique," *IEEE Transaction on Wireless Communications*, vol. 7, no. 2, pp. 532–542, Feb. 2008.
- [98] T. Weingart, D. C. Sicker, and D. Grunwald, "A method for dynamic configuration of a cognitive radio," in *IEEE Workshop on Networking Technologies for Software Defined Radio Networks*, Sep 2006, pp. 93–100.

-
- [99] —, “A predictive model for cognitive radio,” in *Military Communications Conference (MILCOM)*, Oct 2006, pp. 1–7.
 - [100] —, “A statistical method for reconfiguration of cognitive radios,” *IEEE Wireless Communications*, vol. 14, no. 4, pp. 34–40, Aug 2007.
 - [101] E. Adamopoulou, K. Demestichas, and M. Theologou, “Enhanced estimation of configuration capabilities in cognitive radio,” *IEEE Communications Magazine*, vol. 46, no. 4, pp. 56–63, April 2008.
 - [102] P. Demestichas, G. Dimitrakopoulos, K. Tsagkaris, K. Demestichas, and J. Adamopoulou, “Reconfigurations selection in cognitive, beyond 3G, radio infrastructures,” in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (Crown-Com)*, June 2006.
 - [103] E. Adamopoulou, K. Demestichas, P. Demestichas, and M. Theologou, “Enhancing cognitive radio systems with robust reasoning,” *International Journal of Communication Systems*, vol. 21, no. 3, pp. 311–330, March 2008.
 - [104] G. Xu and Y. Lu, “Channel and modulation selection based on support vector machines for cognitive radio,” in *International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, Sept. 2006.
 - [105] G. Dreyfus, J.-M. Martinez, M. Samuelides, M. B. Gordon, F. Badran, S. Thiria, and L. Hérault, *Réseaux de neurones, Méthodologie et applications*, 2nd ed., G. Dreyfus, Ed. Eyrolles, Feb. 2002.
 - [106] B. Le, T. W. Rondeau, D. Maldonado, and C. W. Bostian, “Modulation identification using neural network for cognitive radios,” in *SDR Forum Technical Conference*, Anaheim, CA, 2005.
 - [107] M. K. Simon and M.-S. Alouini, *Digital Communication over Fading Channels*, 2nd ed. Wiley, Jan. 2005.
 - [108] C. Kloeck, H. Jaekel, and F. K. Jondral, “Multi-agent wireless system for dynamic and local combined pricing, allocation and billing,” *Journal of Communications*, vol. 1, no. 1, pp. 48–59, Apr 2006.
 - [109] A. Koutsorodi, E. Adamopoulou, K. Demestichas, and M. Theologou, “User profiling and preference modelling in 4g terminals,” in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Helsinki, Finland, Sept 2006, pp. 1–6.
 - [110] S. U. H. Qureshi, “Adaptive equalization,” *Proceedings of the IEEE*, vol. 73, no. 9, pp. 1349–1387, Sept. 1985.
 - [111] A. Cornuéjols, L. Miclet, and Y. Kodratoff, *Apprentissage artificiel, Concepts et algorithmes*. Eyrolles, Aug 2002.
 - [112] A. Fehske, J. Gaeddert, and J. H. Reed, “A new approach to signal classification using spectral correlation and neural networks,” in *International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Baltimore, MD, USA, Nov 2005, pp. 144–150.
 - [113] A. N. Mody, S. R. Blatt, D. G. Mills, T. P. McElwain, N. B. Thammakhoune, J. D. Niedzwiecki, M. J. Sherman, C. S. Myers, and P. D. Fiore, “Recent advances in cognitive communications,” *IEEE Communications Magazine*, vol. 45, no. 10, pp. 54–61, Oct. 2007.

- [114] H. Hu, J. Song, and Y. Wang, "Signal classification based on spectral correlation analysis and svm in cognitive radio," in *International Conference on Advanced Information Networking and Applications (AINA)*, March 2008, pp. 883–887.
- [115] R. S. Sutton, "Reinforcement learning architectures," in *International Symposium on Neural Information Processing (ISKIT)*, Fukuoka, Japan, 1992.
- [116] L. P. Kaelbling, M. Littman, and A. Moore, "Reinforcement learning : A survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, 1996.
- [117] P. Dayan and C. J. C. H. Watkins, *Encyclopedia of Cognitive Science London*. London, England : MacMillan Press, 2001, ch. Reinforcement learning.
- [118] L. A. Zadeh, "Soft computing and fuzzy logic," *IEEE Software*, vol. 11, no. 6, pp. 48–56, Nov. 1994.
- [119] A. Abraham, *Do Smart Adaptive Systems Exist ?*, ser. Studies in Fuzziness and Soft Computing. Springer Verlag, 2005, vol. 173, ch. Hybrid Intelligent Systems : Evolving Intelligence in Hierarchical Layers, pp. 159–179.
- [120] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing : A Computational Approach to Learning and Machine Intelligence*. Prentice Hall, 1996.
- [121] L. H. Tsoukalas and R. E. Uhrig, *Fuzzy and Neural Approaches in Engineering*. New York : John Wiley and Sons, 1997.
- [122] S. Mitra and Y. Hayashi, "Neuro-fuzzy rule generation : Survey in soft computing framework," *IEEE Transactions on Neural Networks*, vol. 11, no. 3, pp. 748–768, May 2000.
- [123] C.-T. Lin and C. S. G. Lee, "Reinforcement structure/parameter learning for neural-network-based fuzzy logic control systems," *IEEE Transactions on Fuzzy Systems*, vol. 2, no. 1, pp. 46–63, Feb. 1994.
- [124] C.-F. Juang and C.-T. Lin, "An on-line self-constructing neural fuzzy inference network and its applications," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 1, pp. 12–32, Feb. 1998.
- [125] D. Nauck and R. Kruse, "A neuro-fuzzy method to learn fuzzy classification rules from data," *Fuzzy Sets and Systems*, vol. 89, no. 3, pp. 277–288, Aug. 1997.
- [126] S. Nolfi, D. Parisi, and J. L. Elman, "Learning and evolution in neural networks," *Adaptive Behavior*, vol. 3, no. 1, pp. 5–28, Aug. 1994.
- [127] A. Abraham, "Evoanf : a framework for optimization of fuzzy inference systems using neural network learning and evolutionary computation," in *IEEE International Symposium on Intelligent Control*, 2002, pp. 327–332.
- [128] Y.-H. Song and A. T. Johns, "Applications of fuzzy logic in power systems - part 2 : Comparison and integration with expert systems, neural networks and genetic algorithms," *Power Engineering Journal*, pp. 185–190, Aug. 1998.
- [129] N. Kasabov, "Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 31, no. 6, pp. 902–918, Dec. 2001.
- [130] M. J. Middlemiss, "Intelligent information systems for online analysis and modelling of biological data," Master's thesis, University of Otago, Dunedin, New Zealand, Feb. 2001.
- [131] N. Kasabov, *Evolving Connectionist Systems. The Knowledge Engineering Approach*, 2nd ed. New York : Springer, 2007.

-
- [132] N. Kasabov and D. Filev, "Evolving intelligent systems : Methods, learning & applications," in *International Symposium on Evolving Fuzzy Systems*, Sept. 2006, pp. 8–18.
 - [133] A. Kandel, *Fuzzy Expert Systems*. CRC Press, 1992.
 - [134] H. Montazeri and M. R. Meybodi, "A new hybrid model for associative reinforcement learning," in *Proceedings of the Third Information and Knowledge Technology*, Ferdowsi University of Mashad, Mashad, Iran, Nov. 2007.
 - [135] M. A. L. Thathachar, "Stochastic automata and learning systems," *Sadhana*, vol. 15, no. 4 & 5, pp. 263–281, Dec. 1990.
 - [136] K. Najim and G. Oppenheim, "Learning systems : Theory and application," *IEE Proceedings-E*, vol. 138, no. 4, pp. 183–191, July 1991.
 - [137] M. A. L. Thathachar and P. S. Sastry, "Varieties of learning automata : An overview," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 32, no. 6, pp. 711–722, Dec. 2002.
 - [138] M. A. L. Thathachar and V. V. Phansalkar, "Convergence of teams and hierarchies of learning automata in connectionist systems," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 25, no. 11, pp. 1459–1469, Nov. 1995.
 - [139] E. Triantaphyllou, *Multi-Criteria Decision Making Methods : A Comparative Study*, ser. Applied Optimization. Kluwer Academic Publishers, Nov. 2000, vol. 44.
 - [140] S.-J. Chen and C.-L. Hwang, *Fuzzy Multiple Attribute Decision Making : Methods and Applications*, ser. Lecture Notes in Economics and Mathematical Systems. Berlin : Springer-Verlag, 1992, vol. 375.
 - [141] N. Baldo and M. Zorzi, "Fuzzy logic for cross-layer optimization in cognitive radio networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 64–71, Apr 2008.
 - [142] B. Bouchon-Meunier, *La logique floue*, 3rd ed. Que sais-je ? (2702), Presses Universitaires de France, 1999.
 - [143] L. Goh, Q. Song, and N. Kasabov, "A novel feature selection method to improve classification of gene expression data," in *Second Asia-Pacific Bioinformatics Conference (ABPC)*, vol. 26, no. 4, Dunedin, New Zealand, Jan. 2004, pp. 161–166.
 - [144] B. Sklar, "How i learned to love the trellis," *IEEE Signal Processing Magazine*, vol. 20, no. 3, pp. 87 – 102, May 2003.
 - [145] N. Kasabov, Z. Chan, Q. Song, and D. Greer, *Do Smart Adaptive Systems Exist ?*, ser. Studies in Fuzziness and Soft Computing. Springer Verlag, 2005, vol. 173, ch. Evolving Connectionist Systems with Evolutionary Self-Optimisation, pp. 181–202.
 - [146] N. Kasabov, "ECOS : Evolving connectionist systems and the eco learning paradigm," in *International Conference on Neural Information Processing*, Kitakyushu, Japan, Oct. 1998, pp. 1222–1235.
 - [147] M. J. Watts, "Evolving connectionist systems : Characterisation, simplification, formalisation, explanation and optimisation," Ph.D. dissertation, University of Otago, Dunedin, New Zealand, Feb. 2004.
 - [148] V. Vapnik, *Statistical Learning Theory*. New York : Wiley-Interscience, 1998.
 - [149] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121–167, 1998.
 - [150] J. D. R. Joaquim and N. Colson, "Développement d'une chaîne radio numérique reconfigurable," Orange Labs, Stage ingénieur ENSERG, Sept. 2007.

- [151] M. Krondorf, S. Thilakawardana, K. Moessner, X. Wang, A. Gameiro, P. Marques, D. Morche, B. Bochow, M. Dohler, and A. Kountouris, “ORACLE- opportunistic radio communications in unlicensed environments,” White paper, 2007, available online at <http://www.ist-oracle.org> (14 pages).
- [152] A. Wautier, L. Husson, D. Mandic, and I.-O. Plai, “Algorithms for BER-constrained variable-length equalizers driven by channel response knowledge over frequency-selective radio,” in *IEEE Vehicular Technology Conference (VTC)*, Dublin, Ireland, May 2007.